PHD

Designing for Effective Freehand Gestural Interaction

Ren, Gang

Award date:
2013

Awarding institution:
University of Bath

Designing for Effective Freehand Gestural Interaction

submitted by

Gang Ren

for the degree of Doctor of Philosophy

of the

University of Bath

Department of Computer Science

October 2013

Signature of Author ....

Gang Ren

ABSTRACT

Gestural interaction has been investigated as an interaction technique for many years and could potentially deliver more natural and intuitive methods for human computer interaction. As a novel input mode compared to traditional input devices such as keyboard and mouse, researchers have been applying gestural interaction in many different domains. There are many different gestural interaction methods and systems which have been built for both research and mass production. However, most previous gesture user interfaces rely on hand held devices or the wearing of fiducial markers for gesture tracking. The freehand gestures, which are tracked by distance sensors without requiring users to hold or wear special devices, are not fully explored in previous research. Considering that freehand gestural interaction could be easier to use and deployed for ordinary users in every day life, further research should be conducted for designing effective freehand gestural interaction.

In an effort to extend the knowledge and understanding of human factors and interaction design issues relating to freehand gestural interaction, we first provided a review of related works, and analysed the characteristics, design space as well as the design challenges and opportunities of freehand gestural interaction. We then progressively investigated several aspects of freehand gestural interaction including option selection in 2D and 3D layout, object selection in densely populated environments and 3D navigation in public settings. Based on the interaction design, prototype development and user evaluations, we gave the results of user performance, behaviour and preference. We also compared our findings with previous research to extend the state of art. Furthermore, we extended the discussions to a set of practical design suggestions for effective freehand gestural interaction design for different scenarios and interaction tasks. We concluded the directions for the future development of freehand gestural interaction technologies and methods in the end of the thesis.

DECLARATION

At the time of submission, several sections of work from this thesis have previously appeared in peer-reviewed publications. In the following list the full references for these publications are given.

- **Ren, G.**, Li, C., O'Neill, E. and Willis, P.J. (2013) 3D freehand gestural navigation for interactive public displays, IEEE Computer Graphics and Applications 33 (2), 47-55. DOI: http://doi.ieeecomputersociety.org/10.1109/MCG.2013.15
- **Ren, G.** and O'Neill, E. (2013) 3D selection with freehand gesture, Computers and Graphics 37 (3), 101-120. DOI: http://dx.doi.org/10.1016/j.cag.2012.12.006
- **Ren, G.** and O'Neill, E. (2013) Freehand gestural text entry for interactive TV. In: Proceedings of the 11th European Conference on Interactive TV and Video (EuroITV '13). 121-130. June 24-26, 2013. Como, Italy. (Full paper and Demonstration. **Best Paper Award**) DOI: http://doi.acm.org/10.1145/2465958.2465966
- **Ren, G.** and O'Neill, E. (2013) Enhancing 3D Content Consumption on Interactive TV with Freehand Gesture. In: Proceedings of the 1st International Workshop on Interactive Content Consumption at EuroITV 2013. June 24-26, 2013. Como, Italy.
- **Ren, G.** and O'Neill, E. (2013) Freehand Gestural Selection Design for Interactive Television. In: Proceedings of Exploring and Enhancing the User Experience for Television Workshop at CHI 2013. April 27-May 2, 2013. Paris, France.
- **Ren, G.** and O'Neill, E. (2012) Freehand Gestural Marking Menu Selection. In: Adjunct Proceedings of Pervasive 2012. June 18-22, 2012. Newcastle, UK.
- **Ren, G.** and O'Neill, E. (2012) 3D Marking Menu Selection with Freehand Gestures. In: Proceedings of IEEE Symposium on 3D User Interfaces (3DUI '12), 61-68. March 4-5, 2012. Costa Mesa, USA.
  DOI: http://dx.doi.org/10.1109/3DUI.2012.6184185

ACKNOWLEDGEMENTS

I would love to express my sincere gratitude to my supervisor Professor Eamonn O'Neill, for his extraordinary support throughout the past few years. It is his advice and help that have guided me to be a HCI researcher and strive for academic excellence. I would like to express my appreciation to the colleagues and friends in the department of Computer Science at the University of Bath, for their support and help both personally and professionally. I would also like to thank the University of Bath. This research would not have been possible without the funding of an Overseas Research Studentship, as well as the support of excellent facilities and staff members from the University. Finally, special thanks go to my family for their love and support. It is a long journey to complete this PhD and my family have kept me encouraged throughout.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Background and Motivation

In recent years, we have witnessed the huge commercial success of multi-touch interactive products and their widespread adoption for use in many areas. The most compelling characteristics of touch interaction are the directness, fluidity and immediacy. Successful touch sensitive interactive surfaces do not require the user to hold or wear any special devices or markers; they support spontaneous and direct freehand interaction.

However, touch screens are not suitable in some scenarios. For example, users increasingly expect more interactive experiences with public displays or TVs for applications which may include learning, gaming, urban visualization and planning. In such cases, users normally keep a certain distance from the display. As touch screens are normally installed close to or even contiguous with the screen, it is not suitable for these scenarios. Touch on two dimensional space (2D) surfaces is not suitable for interaction in three dimensional spaces (3D) . Although the use of 3D computer graphics is important in a very wide range of applications, user interaction with 3D applications is still challenging and often does not lend itself to established techniques which have been developed primarily for 2D desktop interaction or touch screen.

Meanwhile, many 3D user interfaces now rely on tracking hand-held devices or fiducial markers attached to the user. For example, Wii remote [1], and other hand-held devices can be used for 3D interaction in many areas [155, 111, 49, 62]. Fiducial markers have also been applied in animation and interaction [79, 171, 70]. However, these techniques could be cumbersome or entirely inappropriate in situations in which users do not want to hold or to wear special devices to interact, such as interactive displays in shopping malls or public spaces.

[1] `http://www.nintendo.com/wii`

Thus, extending the fluidity and immediacy of touch screens to freehand gestural user interfaces is a potentially fruitful direction for interaction design. This is especially the case where the 2D desktop devices, touch sensitive interaction techniques or hand-held devices are not suitable. Currently, reasonably accurate 3D sensing techniques which can recognize freehand movements with a single camera are now beginning to be commercially available at low cost (e.g. Microsoft Kinect [2], Asus Xtion [3]). This type of input device does not require on-body attachments or hands-on tracked devices, thus enabling very low configuration interaction. With such input techniques, users can use the interactive system naturally with their hand or body movements and interact directly without complex commands. It also enables interaction with pervasive services and public displays without user set-up or configuration (e.g. wearing or holding specific devices, or connecting their personal devices to the system).

However, there are still many challenges for freehand gestural interaction in which the user's gestures are tracked by a single low-cost camera. Firstly, freehand tracking with a single low-cost camera is currently not as accurate as desktop interaction devices (e.g. mouse), and is sometimes noisy. For example, the accuracy of the Microsoft Kinect depth sensor is about 3mm in the image plane and about 10mm in depth at a distance of 2 meters [180, 135]. Secondly, fine movements such as individual fingers or hand pose, as well as joint orientation cannot be tracked reliably with a single camera from a fixed angle. Thirdly, there is no tactile feedback when the hand is simply moving in the air, no physical device to click or tap, and no physical support for the arm or hand for freehand gestural interaction. Fourthly, it is difficult to design the gesture delimiters for freehand interaction [13]. Thus, many existing methods that require users to wear or hold additional equipment cannot be used directly in freehand gestural interaction.

Freehand gestural interaction is a rapidly developing technology, which could offer rich, natural and immersive user experiences to ordinary users with direct and easy interaction. However, there are many design challenges and questions in this area, such as design for fundamental interaction tasks, as well as users' performance, behaviour and preferences, which are still largely unexplored. We are therefore motivated to investigate potential user interaction techniques to enable the effective freehand gestural interaction.

1.2 Problem Statement and Research Goals

Thanks to the recent development of freehand gestural tracking devices, freehand gestural interaction is now a rapidly developing interaction technique, and shows its potential in many scenarios and settings. However, freehand gestural interaction design remains

[2] `http://www.microsoft.com/en-us/kinectforwindows/`
[3] `http://www.asus.com/Multimedia/Xtion_PRO/`

difficult, and its unique character does not lend itself to existing interaction techniques which have been developed primarily for 2D desktop interaction or touch screen.

Most previous research about gestural interaction is based on touch surface, hand-held devices or fiducial markers attached to the user. There is little work focusing on the the mid-air freehand gestural interaction without holding or attaching the tracking devices, and there is little knowledge about design methods, suggestions or users' behaviour and preference. This creates problems for interaction designers: lack of interaction guidelines and experience may result in a less efficient design process and possibly less useful products. This is especially the case when designers try to simply borrow the design experiences and methods for 2D desktop or touch screen.

So, to achieve the research goal of designing effective freehand gestural interaction, we need to understand the design space of freehand gestural input, as well as its similarities and differences from other gestural input techniques such as touch surface, mouse, hand-hand devices and fiducial markers. Then, based on such understandings and previous research, we should investigate the design for freehand gestural interaction by a series of studies focusing on several important interaction tasks and scenarios. After a series of related design and evaluations, we can then provide the design suggestions and guidelines building upon the studies and findings.

As mentioned before, several main design challenges for freehand interaction include less accuracy, lack of fine movement tracking of fingers, no physical button or surface to click or tap. Thus, our research questions arise as follows:

- What are the characteristics and design challenges of freehand gestural interaction?
- How to design effective freehand interaction techniques?
  - How to address the problem of design of the interaction for gesture delimiters for freehand interaction?
  - How to design the interaction for accurate tasks to enable accurate and relaxed interaction with freehand gesture?
  - How to design the interaction for specific scenarios such as public displays, and what are users preferences and behaviours?
  - What are users' behaviours and preferences when they perform freehand gestures with different freehand gestural interaction designs? What are the similarities and differences among freehand gestural input and other input devices?
- What are the suggestions and guidelines for designing effective freehand gestural interaction?

1.3 Organization of the Thesis

In this section, we presented an overview of the topics and their structure. First, we reviewed existing work on gestural interaction and selection techniques, and analysed the main characteristics and challenges of freehand interaction. In the following three chapters we then reported a series of studies investigating the design of freehand 3D gestural interaction which could be used in everyday applications without complex configurations or attached devices or markers. At the end we summarized the lessons learned, design suggestions and proposed future work and conclusions.

- Chapter 2. The related literature was reviewed in this chapter. We started with the taxonomy and usability of the gestural interaction. And then some more research examples of gestural interaction techniques, such as selection, text input and navigation techniques. At the end of this chapter we gave a detailed analysis of the design space, as well as the challenges and opportunities of the freehand gestural interaction. This chapter builds the ground for the following chapters.
- Chapter 3. As selection is a fundamental interaction task in many applications, this chapter investigates option selection as an exemplar task for freehand gestural interaction. We reported the design and evaluation of two option selection methods. The first study focuses on freehand menu selection in a 2D layout, illustrating how to leverage the successful 2D desktop and touch interaction design in freehand interaction design. The second study explores freehand 3D menu selection, showing how to extend 2D interaction to 3D interaction with freehand gestural interaction. Drawing on the results and findings from these studies, we discussed 3D freehand gestural interaction, proposing some design directions and examples of interaction techniques for effective and usable freehand gestural selection in 2D and 3D environments.
- Chapter 4. Based on the design and findings in Chapter 3, this chapter investigates target selection techniques in dense and occluded 3D environments, suggesting how we might design freehand 3D interfaces by extending lessons from experiences with body-attached or hand-held 3D tracking devices. We report the design and evaluation of two experiments. The first one is target selection in dense and occluded 3D environment, and the second one is text entry with a virtual keyboard. The findings from the design and evaluation of different selection method layouts are reported.
- Chapter 5. In this chapter, we investigated the usage of freehand gestural interaction in public settings and 3D navigation task. We proposed gestural navigation techniques for interactive public displays. We conducted two evaluations, a formal, quantitative lab experiment and an informal, qualitative field study

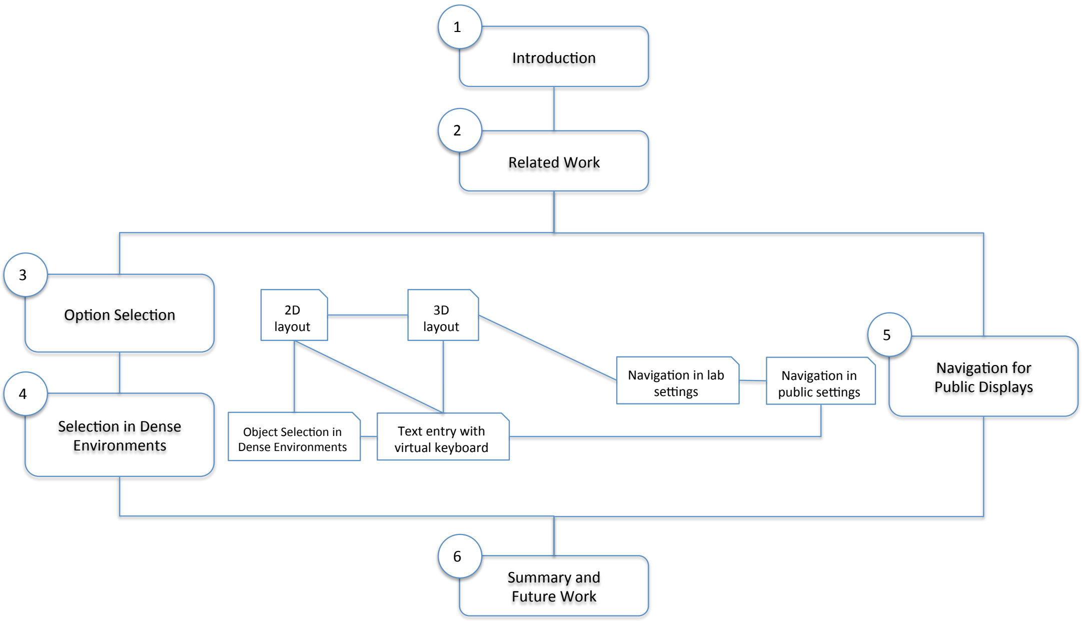


Figure 1-1: *Organization of the thesis.*

to investigate some potential gestural interaction techniques and compared them with traditional desktop interaction techniques using keyboard and mouse. Our results showed no significant difference in navigation performance between freehand gestural and keyboard/mouse navigation techniques. However, the gestural interaction provided a more natural experience in both individual and public environments.

- Chapter 6. We concluded the thesis in this final chapter. We summarized the main findings, design methods and suggestions found in the reported studies, as well as the future work directions of freehand gestural interaction.

1.4 Contributions

In the process of addressing the research problems and going for the research goals, the following contributions are made:

- Analysis of the challenges, opportunities, and design space of freehand gestural interaction based on its features and characteristics.
- A set of design methods, evaluations and suggestions for freehand gestural interaction, including:
  - Menu selection
  - Object selection

  - Text entry
  - Interaction in public settings

- Comparison of freehand gestural interaction and other interaction techniques, such as differences and similarities of design space, task performance and user behaviour.

- A set of general and practical design guidelines and recommendations for freehand gestural interaction designers.

CHAPTER 2

RELATED WORK

2.1 Introduction

Gestural interaction has been investigated for a long time as an interaction technique, and could potentially deliver more natural and intuitive methods for human computer interaction. There are many different gestural interaction methods and systems which have been built for both research and mass production, and these previous work could be a solid foundation of this research. Thus, in this chapter, we provided a brief summary of related work from the perspective of taxonomy and usability of gestural interaction, as well as gestural input techniques, different gestural interaction techniques, and then we summarized the design space and design challenges of freehand gestural interaction in the end of this chapter.

2.2 Taxonomy and Usability of Gestural Interaction

Gestural interaction has been investigated for a long time and many different gesture types have been designed and evaluated, and efforts made to summarize and classify different types of gesture [94, 178, 169]. Karam and Schraefel [94] classified gesture styles as deictic, manipulation, gesticulation, semaphores and sign language. Deictic and manipulative gestures are similar to pointing and manipulating in real life interaction, and they can be used without special learning or training. Gesticulation is the gesturing that accompanies everyday speech so it too requires no special training. In contrast, semaphoric gesture and sign language require a dictionary and even grammatical structures, thus training is necessary before using these gesture types with interactive system. In a review in 2005, Karam and Schraefel [94] found most gesture systems do not focus on a single gesture style (Figure 2-1).

Semaphoric gesture and sign language can be useful in appropriate scenarios, and

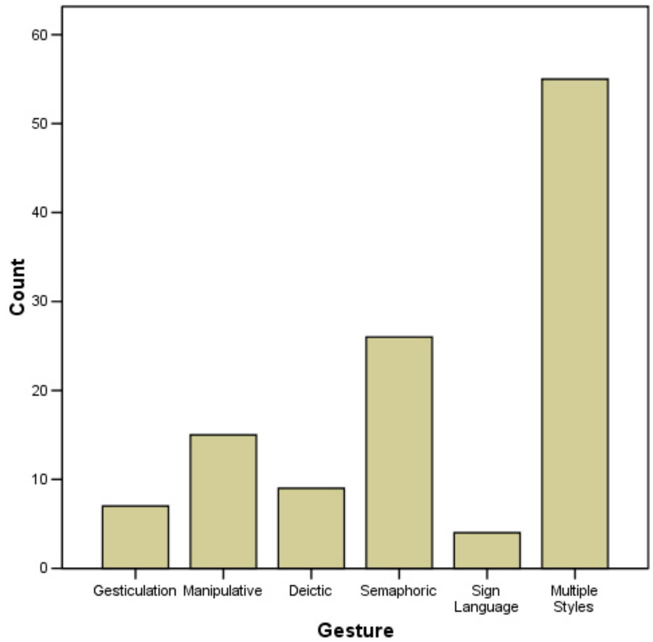


Figure 2-1: *The different gesture styles [94].*

provide powerful tools when they have been thoroughly learned [179, 20]. However, Norman [129] argues that complex gestures are usually not easy to learn and remember, lack critical clues and instant response, and it can be difficult to discover more functions and possibilities. Jetter et al. also argue that we need better model-worlds, not better gestures [89]. Using gesticulation and speech can be an effective means of freehand interaction in many areas, such as virtual object control [23, 16], medical applications [123] and multimedia control [43], however, the introduction of speech could also increase the learning demand and cognitive load.

Most gestures used for target selection or object control are deictic and manipulative gesture. We learn and use these gestures in everyday life. They depend on innate abilities [20], and can be performed directly and simultaneously [13]. Furthermore, the user's gestures can lead to smooth continuous changes of the system state with immediate output [89]. Such directness, fluidity and immediacy are the most compelling characteristics of current 2D touch surface interaction: most existing interaction research for 2D computing provides the appealing features of direct manipulation [20] and walk-up-and-use [13] by using deictic and manipulative gestures.

The directness and immediacy of deictic and manipulative gestures also make them a good basis for freehand gestural interaction, especially for fundamental user interface tasks such as selection. Thus, in the following review, analysis and studies, we focus mainly on deictic and manipulative gestures.

Gestural interaction techniques are widely used in many areas. For example, with Virtual Reality applications, gestural interfaces can be used for navigation[174], selection [115], and many other purposes [32]. Combining gesture and speech could also bring many opportunities for Virtual Reality [23]. Similarly, gestural interaction has also been adopted in Augmented Reality applications for pointing and direct manipulation in both desktop [31] and mobile environments [84], as well as combining with speech as multi-modal interaction [87]. Gestural interaction has also been applied and studied in other areas such as 3D graphics design [194], interactive TV [112] and gaming [109].

To design effective gestural interaction, the physiological characteristics of arm movements may also be considered. For example, the bell-shaped directional tuning curves are recoded [65, 60] when users move their hands in a 2D plane, and the results suggest that a major goal of motor coordination is the smoothest possible movement of the hand. The speed and accuracy of relationships of arm and hand movements have also been widely studied. For example, Fitts' law [57, 159] describes the relationship between movement time, distance, and accuracy for people engaged in rapid aimed movements.

2.3 Gestural Tracking Techniques

In this section we present a brief overview of gestural input techniques from the perspective of enabling input hardware and techniques, focusing on the characteristics of and their affect on the gesture interactions that are possible.

2.3.1 Mouse and Pen Input

The most common input device used on desktop computers today is the computer mouse. It is widely used as a pointing device with graphical user interface (GUI) for menu and object selection [110]. And it can also be used as a gestural input device for many tasks such as menu selection [103] and object manipulations [144].

As a pointing device, pen is also used widely as a gestural input technique. One common used gestural style for mouse and pen is stroke gestures [40], which can be used for creating shortcut with distributed applications [45], video applications [130], photo managements [90], collaborative applications [11], desktop file management and 3D model creation [7].

The usage of pen can also be extended in 3D space. For example, tile menu [166] extends the selection capabilities of pen-based interfaces by using 3D orientation information of pen devices. And finger proximity in 3D space could also been used for desktop applications to enhance user's performance by expanding single and multiple target [188]. Samsung also make this technique commercially available in mobile phone and tablet by S Pen [147], which can detect the pressure applied on the screen, as well as the pen proximity (Figure 2-2).

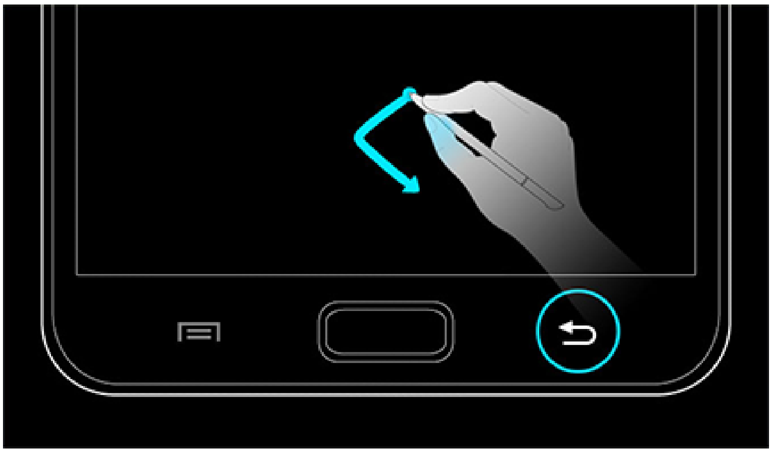


Figure 2-2: *Go back gesture of Samsung S Pen. Draw a less-than arrow while holding down the S Pen button.*

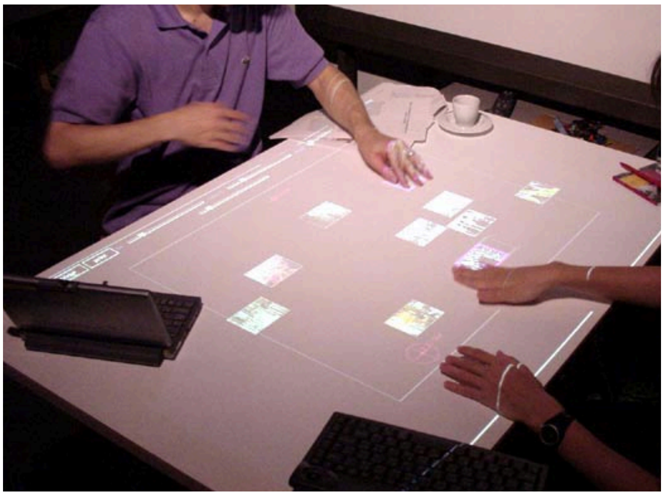


Figure 2-3: *An interactive surface system based on the SmartSkin sensor [139].*

2.3.2 Touch Enabled Surface

Touch sensitive devices are widely used nowadays with the popularity of smart phones and tablets. The touch screen can not only enable similar gesture like mouse and pen, but extending the possibilities with fingers and hands, as well as multi-touch support. For example, SmartSkin [139] can recognize multiple hand positions and shapes, and then calculates the distance between the hand and the surface by using capacitive sensing and a mesh-shaped antenna (Figure 2-3). And multi-finger and whole hand shape are also used with touch sensitive surface for menu selection or manipulation [185].

The touch surface can also be used in different types of devices, from large interactive surface [95], desktop sized screen [54], tablet and mobile phone [138], as well as wearable devices. And combination of such devices in a multiple devices environment

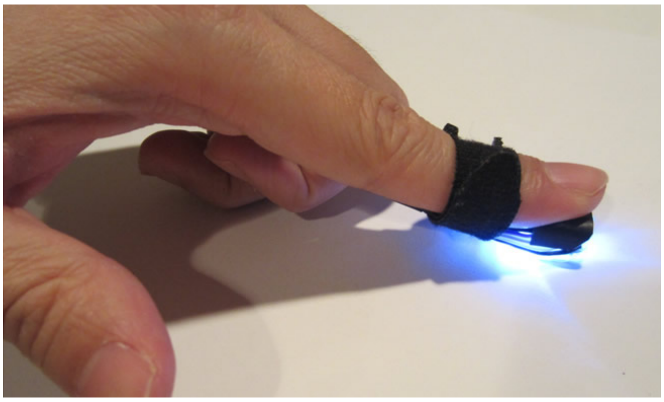


Figure 2-4: *The magic finger device [189].*

[50, 154]. Furthermore, multi-touch surface can not only track finger movements in 2D, but in 3D space as well with augmented depth sensor [163].

It is also possible to apply touch input to all surfaces. For example, magic finger [189] uses a a small device worn on the fingertip to turn any surface to touch enabled device, as shown in Figure 2-4. And Disney Research [148] also proposed a technique enhancing touch interaction on humans, screens, liquids, and everyday objects, which enabled by a technique recognize complex configurations of the human hands and body.

2.3.3 Motion Tracking with Hand-held devices

Hand-held devices are common techniques for motion capture and thus widely used for gestural input in virtual reality and video game applications [109, 153, 26]. For example, there are a number of commercially available implementations of six degree-of-freedom (6-DOF) tracking devices [1] and they have been commonly used for interacting with virtual environments or large displays for directly manipulating the position and orientation of virtual objects [27, 44, 53, 34].

Wii remote [2] is another good example widely used by both consumers and researchers. It is a hand-held device mounted with buttons, and it contains a 3-axis accelerometer, a high-resolution highspeed IR camera, a speaker, a vibration motor, and wireless Bluetooth connectivity. It is used for interaction tasks such as image analysis [62], TV control [156] and text input [92, 155]. Wii remote can also be augmented for better performance [111].

With various sensors on mobile devices, the mobile interaction can also be enhanced by motion tracking, for example, SHRIMP [175] uses camera based motion sensing to enable the user to express preference through a tilting or movement gesture. And to address the problem of gestural delimiter of mobile devices, DroubleFlip [145] is proposed as a unique motion gesture for mobile motion-base interaction. And different

[1] `www.ascension-tech.com`, `www.polhemus.com`
[2] `http://www.nintendo.com/wii`

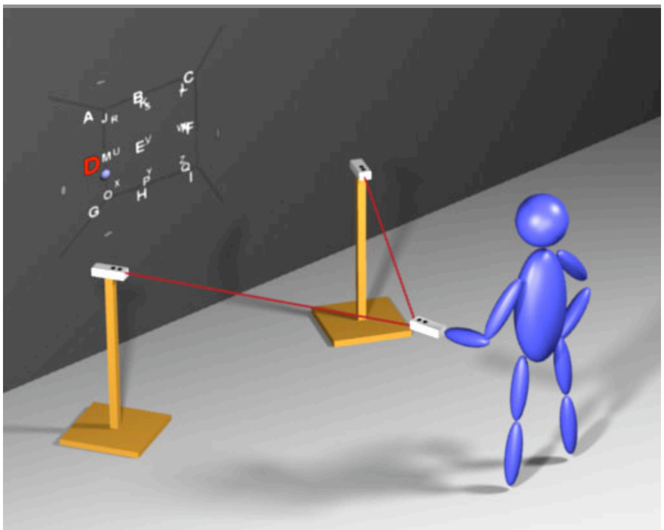


Figure 2-5: *Triangulating position of hand-held Wiimote using 2 fixed Wiimotes on stands for mid-air text input [155].*

motion gesture design methods are also proposed, such as user-defined gesture [146, 184] or design tool [6].

2.3.4 Glove Based Tracking

With a data glove a lot of sensors can be embedded inside, thus small movement can be tracked with accuracy. For example, DataGlove [1] is mounted with flex sensors that can measure finger bending, position and orientation. Thus this type of glove is a typical choice of technology for enabling 3D interaction, such as selection picking, rotating [199]. It is also widely used in virtual environment applications with head mounted displays, such as space research [56], and manipulating virtual objects [162, 176]. Wrist and finger movements can also be used with data gloves for menu selection [127] and text input [126] (Figure 2-6). Other sensors are also possible to mount on gloves. For example, sensors could be mounted on the finger tips so the pinch actions between different fingers can be detected, and such pinch gestures could be used for menu selection [128] and text entry [126].

2.3.5 Computer Vision Based Tracking

There are many investigations using computer vision method to tracking human body and hand. Karam and Schraefel [94, 93] also found computer vision is the most popular enabling method for gestural interaction systems. And computer vision is one of the major technologies for enabling gestures. Computer vision based gesture recognition does not require physical contact with any electronic devices, so the hand can normally move in the air in 3D space without holding or touching anything. And there are two

Figure 2-6: *Glove device used for menu selection [127].*

methods to achieve this, one is attaching some special tracking markers on users' hand or body, and the other method doesn't need such markers so users can just walk in and use.

2.3.5.1 Tracking With Fiducial Markers

Early freehand interaction systems needed fiducial markers on the user to enable tracking of the gestures. This type of computer vision tracking has been used in different tasks and applications, such as skeleton animation [171, 140, 79], virtual reality [119, 19], monitoring users' behaviour [125], interactive environments [12], and pervasive displays [172].

Fiducial markers can work well with a single camera or camera array thus can achieve high tracking accuracy. However, one of the main problem of gestural interaction with computer vision tracking is the absence of physical buttons or surfaces. To address this problem, Vogel and Balakrishnan [173] used the movement of fingers to simulate "mouse-clicking" (Figure 2-7). Design principles and an interaction framework were also developed [172] for interactive public ambient displays with computer vision tracking.

2.3.5.2 Freehand Tracking Without Fiducial Markers

With no fiducial markers, users can interact with the system more directly without the need of wearing markers. These can be achieved by normal RGB cameras. For example, the hand's position and gestures can be tracked in 3D with two cameras [151, 152]. And the fingertip can also be detected by the camera on mobile device [9] or stereo cameras [158].

Besides normal cameras, it is also possible to track freehand gesture with a depth

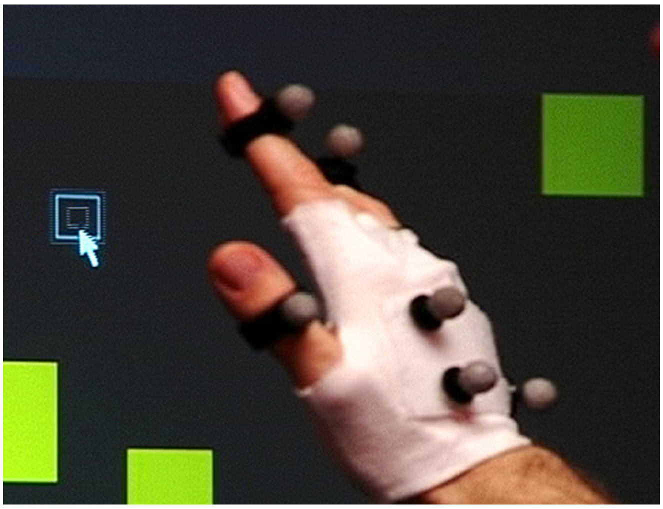


Figure 2-7: *Vicon motion tracking system with passive markers attached to the hand [173].*

Figure 2-8: *Microsoft Kinect sensor.*

cameras, such as Microsoft Kinect [3] (Figure 2-8) or ASUS Xtion [4]. This type of tracking device has the advantage of enabling freehand tracking in 3D space without requiring the user to hold any device in the hand or use fiducial markers. However, such tracking techniques with a single remote camera normally have low resolution and tracking accuracy. For example, the accuracy of the Microsoft Kinect depth sensor is about 3mm in the image plane and about 1cm in depth at a distance of 2 meters [180, 135]. In practice, the skeleton tracking based on the raw depth data can be even noisier.

Research illustrates the directness and immediacy of freehand gestural interaction in daily life use cases. For example, freehand gestural input has been explored using a Kinect sensor for object manipulation [157]. Two handed operation is used to address the lack of hand orientation tracking. Virtual objects can also be manipulated using

[3] `http://www.microsoft.com/en-us/kinectforwindows/`
[4] `http://www.asus.com/Multimedia/Xtion_PRO/`

freehand gestures, including on curved surfaces [13, 14] and projected directly on to everyday objects [74].

The use of freehand interaction has also been compared to other interaction techniques for multimedia control [43]. Participants felt that the freehand pointing is intuitive but needs more precise operation, especially for small target selection such as volume control. Their feedback suggests that, with the improvement of pointing precision, freehand interaction could be a better candidate than the mobile phone as an interface device for remote interaction with large displays.

2.4 Gestural Interaction Techniques

With various gestural tracing techniques, many different gestural interaction methods were designed and evaluated. According to previous studies [28, 94], there are several fundamental tasks for gestural interactive systems, such as selection and navigation. And some other tasks, like text input, are also important for interactive systems in certain scenarios. In this section we summarize interaction techniques in these categories.

2.4.1 Selection

Selection is a topical fundamental task of almost any interactive systems. For example, with graphic user interfaces, a user must be able to specify one or a group of targets before performing further actions. And a user also needs to specify a certain option if she has more than one possible choices. In Windows, Icon, Menu, Pointer (WIMP) interaction framework, which is widely used on many computer systems, the option selection could be achieved by menus. Here we analyze previous research in categories of option selection and object selection.

2.4.1.1 Option Selection

With desktop interactive system, options are normally presented as menus or buttons, which can be selected by mouse or finger tapping on touch screens. Directional gestures have also been used to perform menu selection for multi-touch displays [113] (Figure 2-9). Gestural selection has also been explored in different settings. For example, it has been used to select distant targets on large displays [29].

The linear menu is the most used option selection element in graphical user interfaces. In contrast to linear menus, the principle of a pie menu is to display the menu items around the centre so the selection distance to every menu item is the same. Early work by Callahan et al [33] found that pie menus reduce selection time and errors compared to linear menus.

Marking menus have a similar layout to pie menus with extra supports for expert use, such as hierarchical items and selection simply by moving in a direction with-

Figure 2-9: *The multitouch marking menu [113].*

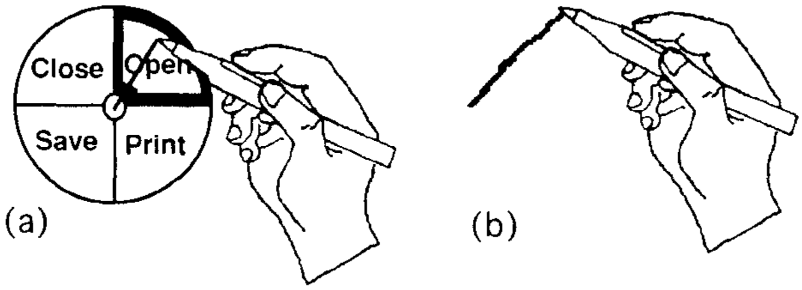


Figure 2-10: *One example of marking menu [106]. (a) Selection using a radial menu and (b) selection by drawing a mark.*

out actually displaying the menu items. Kurtenbach and Buxton [106] (Figure 2-10) showed marking menus to be faster than linear menus, and they found that selection performance with hierarchical marking menus declines when breadth increases to eight or more, or depth increases to two or more [105]. Zhao and Balakrishnan [198] also showed that a simple, i.e. single level, marking menu is more accurate than a hierarchical marking menu, the error increasing with the number of menu items and hierarchical depth, and that less physical input space for selection is needed with a simple marking menu. They also found that for hierarchical marking menus, on-axis directions, e.g. North, South, East, and West, were more accurate than off-axis directions, while no such difference was found with a simple marking menu.

Pie menus and marking menus can be used with different input devices. For example, when using a marking menu with pen devices, the 3D orientation of the pen may be used to aid pen tip selection [166]. Users can also select by holding a mobile phone with a tilt technique [136] based on wrist movement. Wrist and finger movements can

Figure 2-11: *A user is selecting with a menu in a 3D game [41].*

also be used together to select from marking menus using a data glove [127]. Lepinski et al. [113] investigated marking menu selection on a multi-touch surface and found a significant angular error effect for gestures in different directions, for example, small angular error towards Down and large error to Right-Up.

Investigations have also been conducted in different application areas using a single camera (e.g. Microsoft Kinect) and bare hands. Bimanual marking menu selection [71] used a Kinect camera at close range to track the fingers' pose and movement to select from a marking menu. However, this method requires setting up the camera under the desktop at a specific angle, so losing much of the the convenience and benefit of freehand interaction.

Menu selection in 3D games was investigated in [41] using a Wii remote (Figure 2-11). Different menu types including linear, radial and rotary were compared. The results indicated that a radial menu was fastest with and without sub-menus and had fewest errors without sub-menus. Tulip [24] used a 3D menu layout for a wired data glove, displaying the first three menu items on the first three fingers of the user's dominant hand and a "More" menu item on the little finger for triggering additional menu items that are then rendered on the palm. However, evaluation results showed that this method is slower than using a tablet and pen for menu selection.

A survey and taxonomy of 3D menus was provided in [47]; the 3D menus are classified according to various criteria such as intention of use, appearance and structure, or placement. Menus have been used in virtual reality (VR) with different layouts and selection using a wired wand [48]. Evaluation results showed that a pie layout was 10%

faster than vertical lists, and that a fixed menu required more time to select than a contextual menu. 3D menu layout as a cube was evaluated using an in-hand tracker with button clicking, and evaluation results showed that menu items in the central plane could be selected accurately [67].

Various input devices and gestures have been investigated for selection task with interactive TV as well. Bobeth et al. [21] tested freehand menu selection for interactive TV with 4 different designs, and found that freehand gestures could be an appropriate way for older adults to control a TV. A selection task was also investigated in [134], and participants preferred freehand gestural pointing to using a hand-held pointing device. Drawing different shapes in the air can also be used to select objects or menu items [4] with interactive TV, however, certain shapes are not easy to perform and remember, and have low recognition rates. While in another study, stroke gestures has been proved to have some substantial cognitive advantages in learning and recall compared to Keyboard shortcut [5]. User defined gestures for TV were also evaluated in [170]. The results showed that a pointing action was frequently used and a desktop interaction style, such as a push in mid-air to simulate clicking, was observed in many cases.

Another type of selection technique is the target or goal crossing technique [2] (Figure 2-12) in which users need only to cross the target with the cursor, i.e. without needing to point and click inside the target. Option selection with the crossing technique was evaluated [182], and the results showed the crossing technique was faster than point and click, and the motion of both normal and motor impaired user groups was more fluid and stable. Goal crossing has also been used with an area cursor [55]; for example, with the Click-and-Cross cursor the targets covered by the cursor area are arranged as a pie menu around the cursor for a second step of crossing selection. Evaluation results showed that this technique can reduce selection time and corrective submovements for users with motor impairments. The technique of goal crossing has also been used in text entry [183].

Teather and Stuerzlinger [165] tested the performance of the 3D input device in 2D and 3D selection tasks with a mouse or a 3D tracker, and indicated that operation was faster and more accurate when the tracker was used in 2D than 3D.

2.4.1.2 Object Selection

Object selection can include selection in one-dimensional (1D), two-dimensional (2D) and three-dimensional (3D) environments. As object selection task is very fundamental and important, thus it is investigated throughly in previous research. For example, one of the famous study about selection is Fitts' Law [57], which models the relationship between movement time, distance, and accuracy for users performing selection tasks. Fitts' Law has been verified, applied and improved in a wide range of different condi-

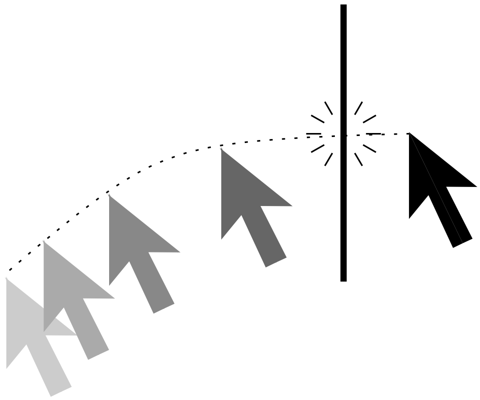


Figure 2-12: *Goal crossing technique [2].*

tions [159]. In this research, as we investigate freehand gestural interaction, with which users can move the hand freely in 3D space, so we are more interested in the design of gestural selection in 3D environments.

In 3D environments, ray-casting is a common approach for selection. It uses a ray or cone to point and select [115]. A common alternative to ray-casting is the hand extension metaphor. In this approach, the user's hand position is tracked in real space, typically using handheld devices or cameras, and the hand position is used to control a cursor in 3D. Ray-casting has been found to have better target selection performance than the hand extension approach [69, 168].

Grossman and Balakrishnan [68] investigated pointing to a single target with the hand extension metaphor in 3D environments with respect to the height, width and depth of the target and the movement angle. They found that moving forwards and backwards to select targets was significantly slower than moving in other directions. They also found that target size along the primary axis of movement has a greater impact on performance than along the other two dimensions, and suggest that designers consider this when they try to reduce the amount of visual space occupied by 3D objects while facilitating selection.

When the target is in a densely populated environment, multiple objects may be in the ray selection range simultaneously and an effective disambiguation method must be provided. For example, to select occluded targets on small 2D touch screens using a finger, Yatani et al [191] rendered the overlapping icons with arrows pointing in a different direction for each object. The user puts her thumb near the target on the screen and moves the thumb in the direction corresponding to the target (Figure 2-13).

An effective extension of ray selection for dense 3D environments is depth ray, in which the user controls a depth marker forward and backward along the ray to

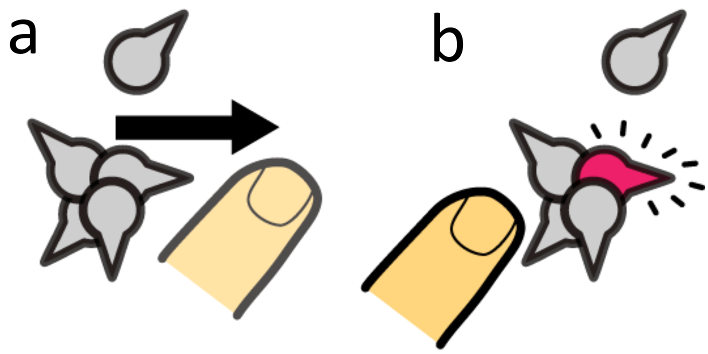


Figure 2-13: *The Escape target selection technique [191]. (a) The user gestures in the direction indicated by the target. (b) The target is selected, despite several nearby distracters.*

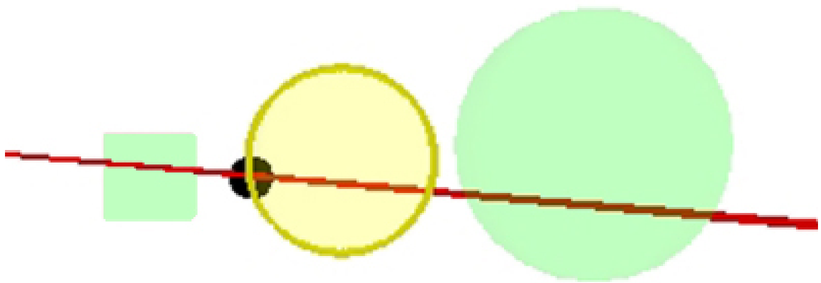


Figure 2-14: *The depth ray selects the intersected target which is closest to the depth marker (side view) [168].*

select a target intersected by the ray. Depth ray integrates the disambiguation and selection phases in one step, and has been shown to be faster than other 2-step selection techniques [69] (e.g. lock ray, flower ray and smart ray) in which the user first selects a group of objects, then disambiguates the target in the second phase. Depth ray has also outperformed other one-step approaches such as point cursor and 3D bubble cursor [168] (Figure 2-14). Multistep selection techniques have also been proposed; for example, Kopper et al. introduced the SQUAD technique [98], using a marking menu to progressively refine selection after a sphere-casting initial selection. Results indicated that SQUAD was much more accurate than ray-casting, and was faster than ray-casting with small targets and in less cluttered environments.

Other techniques have also been developed to disambiguate the target from multiple selectable objects in a 3D space. For example, Schmidt et al [150] proposed several algorithms to select the target by calculating the weights of multiple selected objects, but different algorithms' performance varied in different 3D environments and some algorithms did not work in certain cases, suggesting that it is not easy to find a general algorithm that can handle various 3D environments.

In 3D environments, researchers have also used a spherical menu to display off-screen objects, such as mirror ball [120] and 3D arrow cluster [91]. These techniques can provide 3D location cues for the objects distributed around a 3D environment to support 3D interaction tasks such as navigation.

2.4.2 Text Input

Gestural text input methods have gained more research interest recently and many gestural text entry methods have been proposed and investigated. Most of these methods could also be used with interactive TV. For example, Jones et al [92] used accelerometer-based gesture enabled by a Wii remote and virtual keyboard for text entry. Users achieved 3.7 words per minute (wpm) in first time use and 5.4 wpm after 4 days' practice. A stroke-based text entry method was also designed with data gloves and fiducial markers [126]. Users reached 6.5 wpm without word completion after 2 weeks' practice. A Wii remote was used for text input with large displays in [155]. Three different layouts - circle, Qwerty and 3D cube - were evaluated and the Qwerty layout had the best performance (18.9 wpm), but decreased significantly with more errors as the user moved away from the display. Kristensson and Zhai [101] investigated shape writing recognition to perform word-based text input with a stylus keyboard, and saw high performance in informal trials. However, a test of text entry methods on mobile touch screens showed Qwerty was faster than handwriting and shape writing text entry, and handwriting was the slowest and least accurate text entry technique [37]. Other text entry methods originally designed for stylus and touch screen, such as FlowMenu [72] or Quikwriting [133, 88] could also be used with freehand gestural text input.

Kristensson et al. [100] investigated freehand text entry using free-form alphabetic character recognition, and the evaluation shows a recognition accuracy of 92.7%-96.2%, however, no evaluation on text entry performance is available from their study. Freehand gesture was also used with speech recognition for text entry [83], and 5.18 wpm text input speed was achieved. Although freehand gestural text input with a virtual keyboard is widely used in commercial products (such as games designed for Microsoft Kinect), there is very little research available on this topic. A previous study showed that with the default Xbox 360 gesture based text input interface, the input speed was only 1.83 wpm [83].

Text input with a virtual keyboard is basically a sequence selection of small targets packed together on the keyboard. This is a challenging task for noisy motion tracking with a low-cost single camera system. For example, one typical issue for freehand gesture text input is the difficulty of gesture delimiter design [13]. Accot and Zhai proposed a cross technique which can be used for selection without clicking [2]. A similar method was used for freehand selection in [141], in which users reach towards

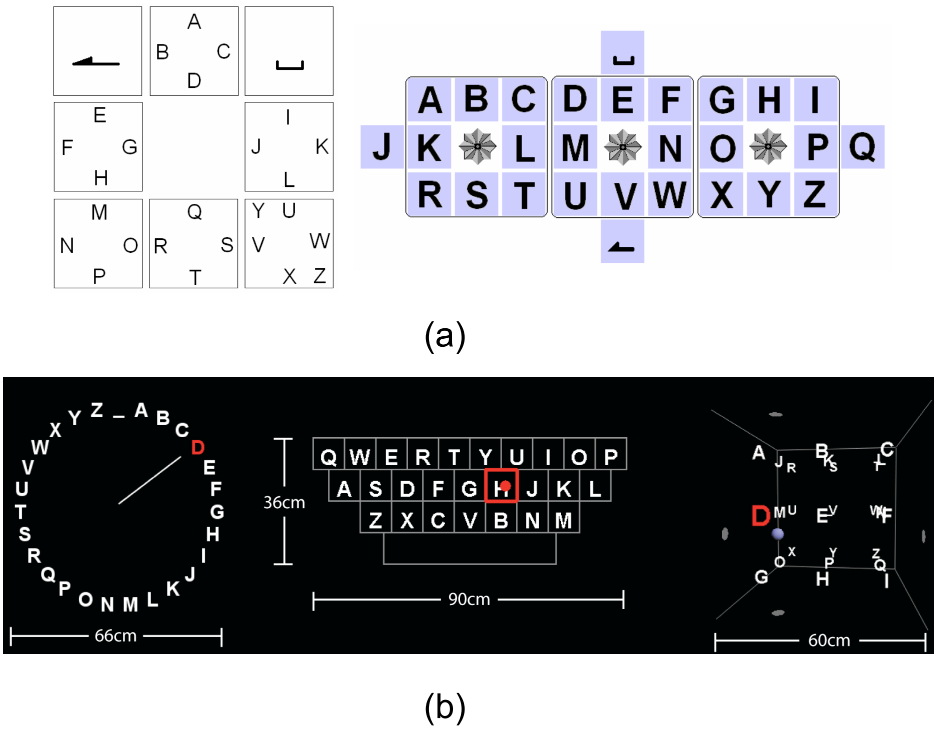


Figure 2-15: *Examples of virtrual keyboard layout. From left to right: (a) matrix-based layout, tri-center layout [92], (b) circle keyboard, qwerty keyboard, cube keyboard [155].*

the target to select without the need to stay inside the target.

Although an “expanding target” can be used to support selection for small targets, such as keys in a virtual keyboard, it can only magnify in visual space but not in motor space when targets are closely packed [122]. A predictor could be used to increase the motor space before the cursor enters the target area, however, the benefit is very limited [122]. And sometimes the expanding target could even have negative effect on the performance. For example, expanding Cirrin [38] try to introduce expanding effect based on the standard Cirrin text input method [118], however, the test results show that that Cirrin with expanding targets is slower and prone to more errors than standard method.

2.4.3 Navigation

Navigation is the task of control the viewpoint, specifying the position and orientation in the interactive environment. Currently, mouse and keyboard are still used as the primary interaction system for navigation task. For example, to navigate in a 3D environment, the non-dominant hand uses keyboard hotkeys to switch between standard interaction modes, such as pan, zoom, and orbit. The dominant hand uses the mouse to control the selected operation. Attempts have been made to combine multiple degrees-of-freedom (DOF) into a single input device, so user effort to control the viewpoint of a 3D model could be reduced.

To overcome the problem of disorientation in a virtual 3D CAD environment, [96] presented an orientation controller called the ViewCube. It can be dragged, or the faces, edges, or corners can be clicked on, to easily orient the scene to the corresponding view. To reduce the occurrence of confusing situations and improve the learning experience, [59] defined seven high-level properties that help users to master navigation tools such as pan, zoom, orbit, look, etc. [66] present the DeskCube, a passive input device to select and control 3D navigation operations that gives users a simple scene-in-hand control experience.

Navigating in larger scenes like a visualised city presents unique challenges. For example, a virtual city encompasses more space than can be viewed from a single vantage point, so users often have to switch between different views to obtain the optimal navigation experience. To address this problem, [164] presented a technique that couples control of movement speed to camera height and tilt, allowing users to seamlessly transition between local environment views and global overviews. [121] presented a suite of tools that automatically sense the size of the environment and adjust the viewing and travel parameters accordingly through simple mouse-based controls.

However, the user experience of navigating a virtual 3D environment is usually less satisfactory using a desktop setting, due to the small display range and unnatural interaction provided by mouse and keyboard operation. Immersive virtual reality (IVR) systems have become more and more popular for navigating virtual scenes, due to their ability to let the user be "inside" the model and to interact directly with the model using physical interactions. *CAVE* [46] for the first time achieved the goal of producing a large angle of view with high-resolution full-color images, allowing a multi-person presentation format. Later improvements included using use real-time computer vision techniques [137] to capture real world surface geometry and reflectance information so that one can project images on the surfaces, render images of the surfaces, or interpret changes in the surfaces.

Efforts have also been made to improve the physical interaction techniques, such as using movements of the eyes, head or body instead of traditional mouse and keyboard navigation. For example, Doulis et al. [51] used a double short stick (NaviStick) and single long stick (Kwisath) tracked using 6DOF tracking to navigate in a virtual environment. Turning and pitch were controlled by rotating the stick. Similar devices such as a 3DOF broom stick [30] and Wii remote [49] have also been used to control 3D navigation. The relationship between display size, user performance time, amount of physical navigation, and amount of virtual navigation were studied in [10]. Their results suggested that larger displays offered improved user performance. Frees and Kessler [61] present an interaction technique which draws on the user's behaviour in the environment to determine whether he has precise or imprecise goals in mind. For

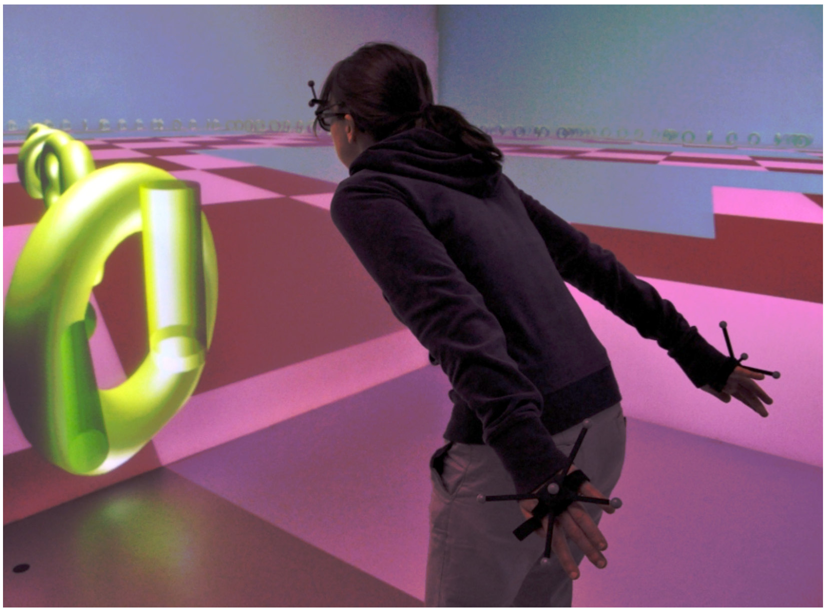


Figure 2-16: *A bimanual body-directed travel technique. The standing user's head and hands are tracked [174].*

instance, when precision is desired, the system dynamically adjusts the relationship between physical hand movements and the motion of the controlled virtual object, making it less sensitive to the user's hand movement. And inside a CAVE display, 3D navigation based on gestures and motion tracking is also used. For example, Kapri et al. [174] used a bimanual body directed navigation technique, in which the user's hands and body are tracked by infrared optical tracking markers (Figure 2-16). Users can also walk inside the 4-sided displays to navigate in the large virtual environments [42, 86].

2.5 Freehand Gestural Interaction Design Space and Challenges

There is considerable previous gestural interaction research focused on 2D multi-touch gestural interfaces and, separately, on 3D gestural interaction enabled by wearing or holding some device. In seeking to understand the extent to which the fluidity, immediacy and physicality of 2D surface interfaces can be extended to work effectively and usably with freehand 3D gestural interaction, we must be aware of the similarities and differences between 2D surface interaction, 3D interaction with attached/held devices, and freehand 3D interaction enabled by remote tracking of the user's bare hands.

In this section, before discussing the design and evaluation of freehand interaction for specific tasks, we first discuss some general challenges of freehand interaction. Gestural interaction techniques enabled by fiducial markers share some usability considerations with freehand interaction which has no need of markers, however, they are very different in many aspects such as tracking accuracy/robustness, difficulty of system setup for daily use, and interactivity cost associated with extra requirements to wear markers. In this thesis, we use the term freehand interaction mainly for interaction using the bare hands tracked by a single remote camera, without wearing any markers or holding any devices in the hand.

2.5.1 Freehand Gestural Interaction Design Space

Different input devices can strongly affect performance on the same task. For example, Kurtenbach and Buxton [107] found that different devices gave markedly different performance with marking menu selection. They found, for example, that the combination of a track-ball and ink trail was not a good design since the ink trail could be messy when using the track-ball to select. Furthermore, a stylus could make better straight strokes compared to a mouse, and it also outperformed the mouse in other gestural interaction tasks such as drawing shapes.

Freehand gesture tracked by a low cost camera is a very different input technique compared to desktop input devices and 2D touch surfaces. Some interaction designs, such as gesturing in the air for selection and manipulation, may be easy to learn and work well for freehand interaction [157], however, directly copying established 2D user interface techniques, such as using a hand open/close to simulate a button click, may not be a suitable design for freehand interaction. Here we analyze the characteristics and challenges of freehand interaction tracked by a single low cost camera, comparing it with other input techniques.

2.5.1.1 Similarities

Freehand 3D interaction shares the spontaneous and direct nature of interaction with 2D touchscreen devices, in which users can use their own hands to interact directly with the touch-sensitive surfaces and the freehand interactive system. This walk-up-and-use [13] interaction style, in which no wearing or holding tracking devices is required to interact with the system, can lower the interactivity cost, thereby helping ordinary people to use the system in their daily lives.

3D interaction with tracking enabled by attaching or holding input devices also has many common features with freehand 3D interaction. Both of them enable the use of 3D interfaces with 3D motion tracking in mid-air. Users can move their hands or body freely in 3D space to control the system. Thus, freehand 3D interaction can leverage some of the experiences and lessons from previous designs and evaluations of 3D interfaces.

2.5.1.2 Differences

There are still many differences between 3D freehand interaction, 2D touchscreen interaction, and 3D interaction through attached or hand-held input devices. With a 2D touch sensitive surface, users can put their fingers on the surface and there is negligible distance between the finger and the virtual object: they both appear on the same 2D surface. In contrast, 3D interaction normally requires the hand to be moving in the air, often with some distance between the hand and the target on the display.

Hence, with 2D interactive systems, given that the pointing position can be tracked accurately with a mouse or touch screen, the user's performance and behavior is influenced more by the size and location of the targets on the surface. For example, small targets on a mobile phone screen require small finger movements [191], while a large, fixed touch surface encourages large hand movements and more fingers on the display [113], or even multiple users operating at the same time [80].

On the other hand, for 3D interaction, although the size and location are still important factors, the tracking device's accuracy plays a more important role for user performance and behavior compared to 2D interactive systems. For example, with a data glove or trackers attached to the fingers, individual finger movements can be tracked and used [24, 126, 70]; with a hand-held device such as a Wii remote or mobile phone, wrist movement and rotation can be tracked and used [92, 155, 136]. So users can keep their hand movements within a small range and use the fingers and wrist with these high accuracy tracking systems. However, this is not the case for freehand gestures tracked by a single, low resolution camera. For example, the accuracy of the Microsoft Kinect depth sensor is about 3mm in the image plane and about 1cm in depth at a distance of 2 meters [180, 135], and the accuracy decreases with distance. Furthermore, most recognition software based on such depth sensors (e.g. Microsoft Kinect for Windows) can only track hand movements reliably, not finger or wrist movements.

So with freehand interaction in 3D using such devices, the user currently cannot leverage the movement and rotation of fingers and wrists, and a larger hand movement range is normally required because of the low tracking resolution. Two handed operation is also sometimes needed to address the lack of orientation tracking [157, 15]. This means that freehand tracking may require more physical demand and effort compared to existing interaction techniques. So a simple and easy to use interaction design is even more important for freehand interaction.

Other differences include the full 3D movement range of freehand interaction and the corresponding lack of physical support for the hand and fingers. For 2D surface computing, the touch input with fingers or stylus is normally constrained and supported by the 2D physical surface. In contrast, with freehand 3D interaction, the hand is moving freely in the air. Previous studies suggest that compared to hand movement in 3D, the physical support of a fixed 2D plane can result in better performance, and mapping user movements from 3D to 2D could produce more accurate operation [165]. Thus, the designer of 3D freehand interaction should carefully consider the impact of full 3D freedom of movement. Furthermore, friction on the physical surface can also make the movement and momentum of the fingers more gradually and smoothly come to a halt [129], which again is not the case for freehand movement.

2.5.2 Challenges

As noted above, freehand tracking with a single camera is not as accurate as desktop devices such as a mouse, or 3D tracking devices attached to or held in the hand. The resolution is relatively low and the tracked hand position can be noisy. Considering also that the hand is normally moving in the air without any physical support, and making relatively large motor movements, the input data can be quite low in accuracy, and there may in addition be some arm fatigue issues.

With a 2D touch sensitive surface, the user can put her hand very close to the virtual object on the screen. In contrast, with freehand interaction, users normally need to keep a certain distance from the display, making it more like a remote control rather than the direct control of a 2D surface interface. There is also no tactile feedback from physical devices in freehand interaction. Thus, alternative sources of system feedback, for example visual or auditory feedback, may be desirable to complement freehand interaction.

Benko [13] listed several challenges for freehand gestural interaction in 3D space, including gesture tracing, design and learning. Further criticisms from Norman [129] included several drawbacks of gestural interfaces, such as not being easy to learn or remember, and a lack of critical clues for interaction or discovery. Bowman [25] also suggest increasing the level of specificity relative to particular input devices as a technique designed for one input device may exhibit serious usability problems with another.

With the new freehand tracking techniques enabled by a single low cost camera, 3D freehand interaction can be used by ordinary users in daily life without the costs, in many senses, of using tracked physical devices. However, this approach also means more design challenges, such as lack of accurate tracking, no reliable fine movement tracking and pose detection, no tactile input or feedback from physical devices, and the lack of physical support from a 2D surface. Below we summarise the main design challenges for 3D freehand interaction.

2.5.2.1 Gesture Delimiters

A common issue with freehand interaction is the gesture delimiter [13]. Hand held devices are typically mounted with buttons, therefore a button press or release can be used as a delimiter or trigger. With 2D touch surfaces, contact with the surface provides straightforward delimiters. Users touch the surface to interact, and lift off to end the action. However, when users move their hands freely in the air, it is more difficult to determine the beginning and end of a gesture, distinguishing a given gesture from other gestures or just from casual movements.

Several different gesture delimiters have been designed for gestural interaction, such as finger movements [173, 70], pinch posture [16, 70, 181, 71, 126], and hand posture [157]. Most of these previous designs require fine tracking of finger movement: some designs require attached trackers [173, 70] or data gloves [24, 126], while the freehand designs are normally sensitive to camera viewpoint and distance [157, 71].

For freehand interaction tracked with a single inexpensive camera, the large joints of the body can be tracked effectively most of the time, whereas the orientation of the small joints, finger movement and posture cannot be tracked reliably. Thus, for tasks like selection, gesture recognition can be robust for various settings if the gesture delimiters are based mainly on hand movement.

2.5.2.2 High Precision Interaction

The low resolution and noisy input make high precision interaction a big challenge for freehand interaction. A similar issue also exists in 2D touchscreen interaction with small displays. For example, the finger touch area could be too big for precise pointing. Some studies have been performed to investigate this issue on a small touch screen with one finger [191] and on a large screen with two handed interaction [17].

For freehand interaction, the conflict between the requirement for precision and the lack of accurate input can be even stronger. As hand movements in the air are often larger than on a 2D desktop interface or touch surface, the interactivity cost of each action or task is also larger. Thus, an error-prone design can lead to an extremely unusable freehand user interface in which users wave their hands in increasing frustration to recover from errors, only to make new ones.

Again, combining the new possibilities provided by 3D freehand interaction with appropriate lessons learned from existing 2D surface computing research and practice, high precision interaction is still possible with 3D freehand interaction.

2.5.3 Opportunities

Apart from challenges listed in previous section, freehand gestural interaction also brings many benefits and opportunities. The first and most important one is “walk up and use” as described by Benko in [13]. It enables multiple users to interact directly and simultaneously. And there is no need of wearing or holding special devices as their bodies are the input. Thus, groups of users can engage the interaction without needing to take turns or learn complex commands. They can also interact at a distance from the display without blocking the screen, so other audiences can share the interactive content at the same time. This benefit will hugely enhance the development of interaction in public space, living rooms or conference rooms, where multiple users could take part in the interactive applications for entertainment, education or communication.

Another opportunity brought by freehand gestural interaction is it extends the interaction to 3D dimension. Most traditional desktop interaction techniques such as mouse and touch screen limit users in a 2D surface. So when users need to perform 3D interaction tasks, they have to map their operations from 2D to 3D, which is indirect and unnatural as the 2D input space does not match the 3D output. Freehand gestural interaction, on the other hand, naturally happens in 3D space when users move their hands in the air. So it could potentially enable more natural and direct 3D interaction.

2.6 Summary

In this section, we presented a literature review of gestural interaction techniques. We gave a brief analysis of related work about taxonomy and usability of gestural interaction, gestural input techniques, different gestural interaction techniques. And we presented a summary of the design space by comparing the similarities and differences with other gestural interaction techniques. We then identified two design challenges of and design opportunities of gesture delimiters and high precision interaction, as well as opportunities provided by freehand gestural interaction such as interactive public displays and 3D interaction.

Many years of research in gestural interaction suggest that it could be a novel interaction technique offering natural and enhanced user experience. However, there is still very little knowledge focused on freehand gestural interaction for both research community and product designers. Building upon the analysis of previous work, the following chapters of this thesis progressively investigate several aspects of freehand gestural interaction.

CHAPTER 3

FREEHAND GESTURAL OPTION SELECTION

3.1 Introduction

Selection is a fundamental interaction task in many applications. However, although considerable research has been conducted on selection techniques, it is still a challenge to select targets using freehand gestures. In this chapter, we investigated the option selection as an exemplar task for freehand gestural interaction. In an option selection task, users try to select one target from a set of predefined options (normally referred as "menu" in "windows, icons, menus, pointer" metaphor). As the size, number and layout of the options can be controlled, an option selection task serves as a good start point for freehand gestural interaction design.

In this chapter, we report a set of 2 related evaluation studies about the analyse and design of freehand option selection in 2D and 3D layout. In the first study, we focused on the selection in 2D layout, designed two freehand selection techniques and conducted an experiment to compare their performance with and without continuous visual feedback. And then we extended the design to 3D space, reporting the adapted design of selection techniques and layout in 3D, as well as the user evaluation and results. We also provided some interesting findings and discussions by comparing our studies and previous research to reveal the characteristics of freehand gestural interaction. Finally we also reported some practical guidelines based on our findings.

3.2 Option Selection in 2D Layout

Option selection in 2D layout is widely investigated with settings of desktop and touch sensitive surface. However, very little research and knowledge is available for the 2D option selection with freehand gestural interaction, which is a very basic interaction task. In this section, we provided design and experiment focus on freehand option in 2D layout. We presented two selection techniques, Stroke and Reach, that aid option selection using only freehand gestures. Users could perform a directional gesture to select the menu items, and the gesture is recognized based on hand movement direction for Stroke, and hand location for Reach. Both techniques support selection with and without continuous visual feedback of hand position. We also performed an experimental evaluation investigate the two selection techniques, the effect of visual feedback of the user's hand position and different target directions.

3.2.1 Freehand Option Selection Design

3.2.1.1 Selection Gesture

For freehand gesture interaction without holding or wearing any device or fiducial marker, we can fairly easily track joint positions such as hand, elbow or shoulder, however, small movements like wrist rotation or finger movement are difficult to track precisely. Given our requirement for reasonably low cost pervasive technology that can perform well in potentially visually (and otherwise) noisy environments, small movements of the finger or wrist are therefore not suitable for current freehand tracking technology. Hence, we prefer to use the acceleration or movement of the hand, or the angle from hand to elbow or shoulder for tracking freehand gestures.

For option selection, after indicating the target, another action is typically required to confirm the selection with marking menus, such as button clicking [136, 98], pen down/lift [197] or finger movement [127]. However, button clicking is not possible in freehand gestural interaction due to the absence of a handheld device. Similarly, pen down/lift on a touch surface is not suitable either because there is no such surface available in freehand gestural interaction. Escape [191] and cross selection [2] have shown that it is possible to perform gestural selection without an explicit confirmation action. Drawing on previous research, we designed two option selection techniques to enable selection and confirmation with a single directional gesture.

In Escape [191], the direction of finger movement is used to select the target. We applied the same strategy to freehand gesture selection (Figure 3-1a). The user moves her hand in the appropriate direction, the direction is detected, and the menu item in the corresponding direction is selected. Given that freehand motion tracking may be noisy and less accurate than pointing on a touch screen, we added a speed requirement: the directional gesture end position is determined when the hand movement speed is faster than a threshold. The direction is then calculated using this end position and the previous hand position (e.g. 0.1s before). We call this technique Stroke.

Besides using hand movement speed, we can also use hand movement position to define a gestural selection. On desktop interfaces, the most common selection technique is to point to the target and then confirm the selection with mouse click, or tap the target with finger or stylus on a touch screen. Goal crossing [2] is an alternative selection technique in which, rather than entering the target area to click or tap, the user simply moves the cursor or stylus across the target boundary to select.

Accot and Zhai [2] showed that crossing performance is better than pointing for large and proximate targets. Research has also shown that goal crossing is a promising technique for users with motion impairments [182, 55]. Two major challenges for people with motor impairments are the difficulties of positioning the cursor accurately within the target area and executing a click [182], which are quite similar to the challenges of 3D freehand interaction, as described above. Since goal crossing has been shown to mitigate these two problems, it could also be a candidate selection technique for 3D freehand interaction.

Therefore, in addition to the Stroke technique based on hand movement speed in a certain direction, we designed and evaluated an alternative selection technique for freehand gesture, in which the target is selected when the hand position crosses the corresponding target border (Figure 3-1a). We call this technique Reach.

3.2.1.2 Display Design

Radial menu layout is proved to have better performance than linear layout [33, 41], thus we design the freehand option selection mainly based on radial layout. There are also various types of option display layout, such as pie [33], rectangular zone or polygon [197]. We used the polygon style display in our freehand gestural selection design because it can support both the Reach and Stroke techniques.

We chose the number of menu items for freehand selection based on previous work. With pie cursor [58], four and eight items were tested, and Escape [191] used 8 directions for its icon arrows. More items have been tried in [26], and results showed slower selection time and higher error rates for more than eight menu items. Zhao et al. [197] also conclude that to maintain acceptable accuracy rates, the marking menus should not exceed 8 items. Given that the movement ranges of a thumb [191], mouse [33], or handheld tracking device [69] are smaller than an arm, the radius of the menu and distance between each menu item could be larger in our freehand gestural selection setting. This makes our menu potentially able to handle more menu items. Taking all these constraints into account, we chose 8 as the number options in the menu.

3.2.2 Experimental Evaluation

We conducted a controlled experiment to investigate the two selection techniques. Furthermore, we also investigated another two important elements in gestural marking menu selection: continuous visual feedback of the hand gesture and target item direction.

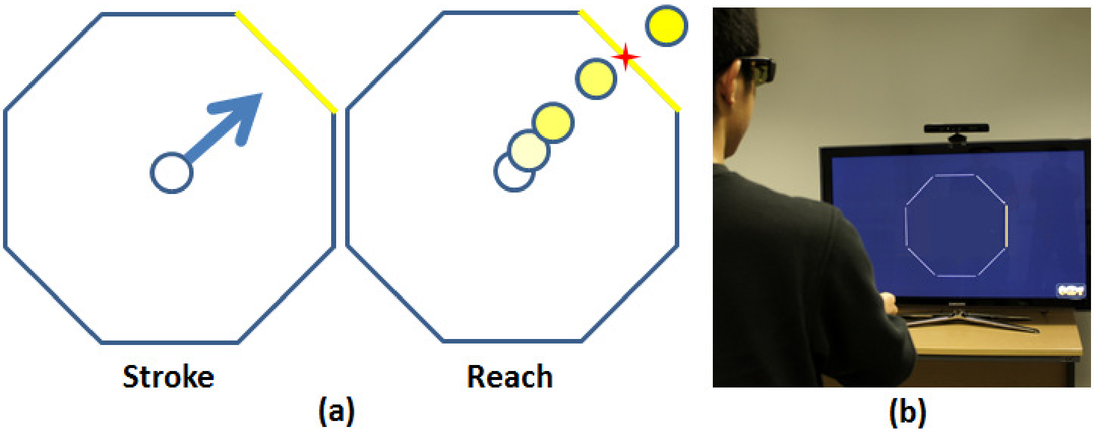


Figure 3-1: *(a) Stroke and Reach selection technique (b) Experiment settings*

3.2.2.1 Equipment and Setting

The 3D environment was displayed on a Samsung PS50C680 50" 3D plasma TV with 120 Hz refresh rate at 1280x720 resolution. Samsung SSG-2100AB/XL 3D active glasses were used by participants to view the display in stereoscopic 3D. The height from the ground to the centre of the display was 107 cm, and the user stood 200 cm in front of the monitor. The user's movements were tracked using a Microsoft Kinect camera, with a refresh rate of 30 fps, and the OpenNI API on Windows 7. The Kinect camera was placed on top of the display, as shown in Figure 3-1b.

3.2.2.2 Independent Variables

The independent variables were Technique (Reach, Stroke), continuous visual Feedback (with feedback, no feedback) and TargetDirection (Up, Down, Left, Right, Left-Up, Right-Up, Left-Down, Right-Down).

3.2.2.3 Hypotheses

- There will be significant differences in selection time and error rate when participants select with Reach vs with Stroke.
- There will be significant differences in selection time and error rate when participants select with and without continuous visual Feedback.
- There will be significant differences in selection time and error rate when participants select in different directions.

3.2.2.4 Participants

We recruited 16 people (12 male and 4 female) ranging between 22 and 30 years of age with a mean age of 25.25 ($sd = 2.41$). All participants were postgraduate students and were paid 15 GBP each. They were all right-handed and had some experience of gestural interaction, such as using a Wii remote or Microsoft Kinect for games.

3.2.2.5 Procedure

To invoke the 2D menu, we introduce the pull gesture, i.e. moving the hand backwards. We use this gesture because menu item selection gestures are performed in the 2D plane perpendicular to this pull gesture direction, so the pull gesture to invoke the menu can be easily separated from the gesture to select menu items. The speed for detecting the gestures was determined in pilot studies.

Each trial started with a pull gesture by the right hand. The pull gesture is detected when the hand moves faster than a speed threshold (0.5 m/s). After the pull gesture was performed, a marking menu presented in an octagon shape appeared with the target edge highlighted in yellow. Each edge was 24 cm from the octagon's center. In the continuous visual feedback conditions, the user's hand position was represented as a 2 cm diameter yellow sphere starting in the center of the octagon, which then continuously followed the user's right hand movement. In the no feedback conditions, the sphere did not appear.

With the Stroke technique, the hand movement direction is detected when hand movement speed is faster than 0.4 m/s, and the item in the corresponding direction is then selected. With the Reach technique, the user has to move her hand in the target's direction for at least 24 cm to go through the target edge to select. The selected edge disappears when a selection is detected. An error sound is played if the user selects an item other than the target. If the target is selected, the user is instructed by text on the display and an audible beep to start the next trial.

3.2.2.6 Design

A repeated measures within-participants design was used. There were 4 sessions, 1 for each combination of Technique and Feedback. In each session, there was a practice block of 24 trials (3 trials in each direction) and a test block with 56 trials (7 trials in each direction). Sessions with the same Technique were contiguous, giving 8 orders of the 4 sessions, counterbalanced across two groups of 8 participants. After the test, the participant answered a questionnaire on technique, feedback and direction preferences. The whole session took about 45 minutes. Thus for each participant, 320 trials were performed in total.

3.2.3 Results

The selection time, error rate, hand movement in 3D and user preference were recorded and the collected data are analysed here.

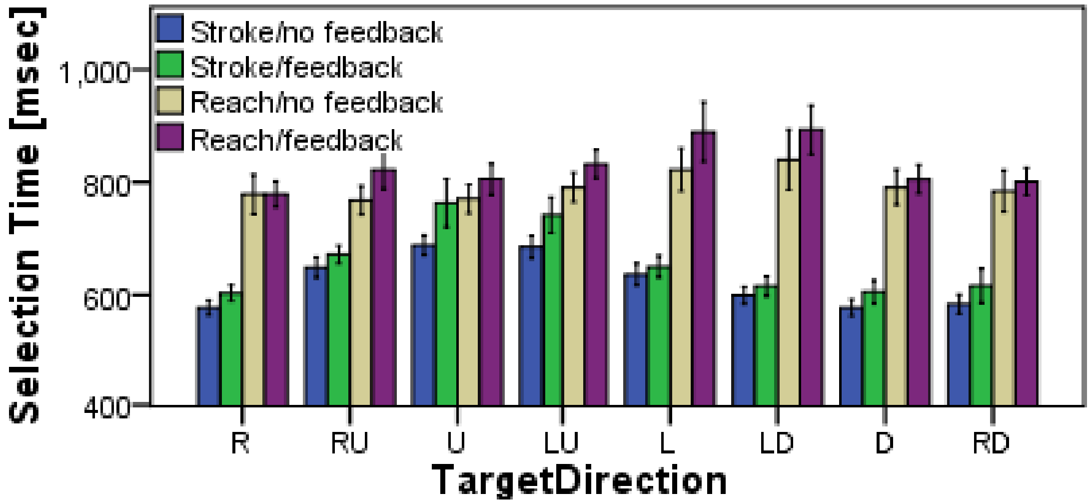


Figure 3-2: *Mean selection time (In this and all later charts, error bars represent 95% confidence intervals).*

3.2.3.1 Selection Time

A repeated-measures ANOVA for Technique x Feedback x TargetDirection was performed. Main effects were found for Technique ($F_{1,15}$= 66.08, $p < .001$), Feedback ($F_{1,15} = 4.59, p < .05$), and TargetDirection ($F_{7,105} = 16.59, p < .001$). An interaction effect was found for Technique x TargetDirection [1] ($F_{2.73,41.00} = 13.82, p < .001$). Mean selection times for Stroke were 0.63s without feedback and 0.65s with. For Reach they were 0.79s without feedback and 0.82s with. Post hoc Bonferroni pairwise comparisons showed that Stroke was significantly faster than Reach ($p < .001$), and selection time was significantly slower with feedback than without ($p < .05$), as illustrated in Figure 3-2.

Further analyses were performed using a one-way repeated-measures ANOVA for TargetDirection using Stroke and Reach. With Stroke, each of Right, Down, Right-Down and Left-Down were significantly faster than all of Up, Right-Up and Left-Up, while with Reach, each of Right, Right-Up and Right-Down were significantly faster than Left-Down ($p < .05$ or more significant for all pairs). In Figure 3-3, blue directions were significantly faster than green directions ($p < .05$).

3.2.3.2 Error rate

Users were required to select the target successfully, if necessary trying more than once. Each failure to select the correct target was recorded as an error and the erroneous direction was also recorded. A repeated-measures ANOVA was used for Technique x Feedback x TargetDirection. We found main effects for Technique ($F_{1,15} = 35.37, p <$

[1] The sphericity assumption was not met so the Greenhouse-Geisser correction was applied; the corrected degrees of freedom are shown.

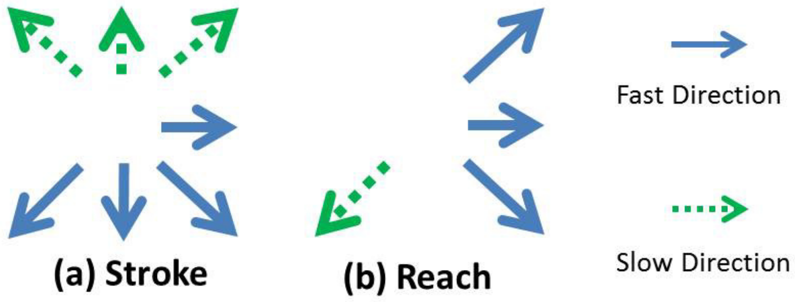


Figure 3-3: *Selection time comparison in different TargetDirections*

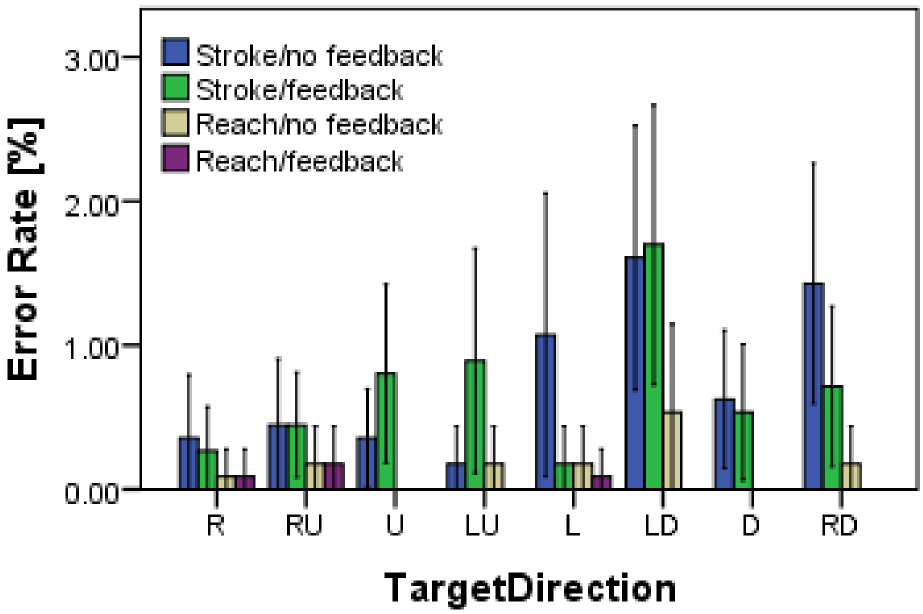


Figure 3-4: *Error rate for Technique x VisualFeedback x TargetDirection*

.001), and TargetDirection [2] ($F_{3.39,50.92} = 4.59, p < .01$), but no significant effect was found for Feedback ($F_{1,15} = 2.11, p = .17$). We found an interaction effect for Technique x TargetDirection ($F_{7,105} = 3.17, p < .01$). Mean error rates for Stroke were 7.56% without feedback and 6.88% with. For Reach they were 1.98% without feedback and 0.55% with (Figure 3-4).

We also compared errors between the on-axis directions (Right, Up, Left, Down) and off-axis directions (Right-Up, Right-Down, Left-Up, Left-Down). Analysis using a two-tailed dependent T-test found that whether the direction was on-axis or off-axis had a significant effect on error rate for Stroke ($t_{15} = -20.72, p < .001$) and for Reach ($t_{15} = -4.07, p < .01$). On-axis directions were more accurate than off-axis directions using both techniques (Figure 3-5).

[2] The sphericity assumption was not met so the Greenhouse-Geisser correction was applied; the corrected degrees of freedom are shown.

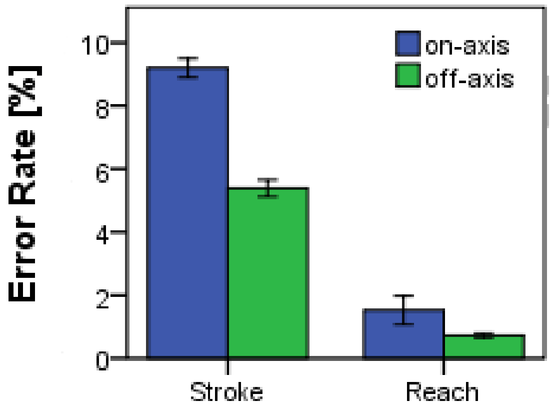


Figure 3-5: *The error rate for on-axis and off-axis TargetDirections and selection techniques*

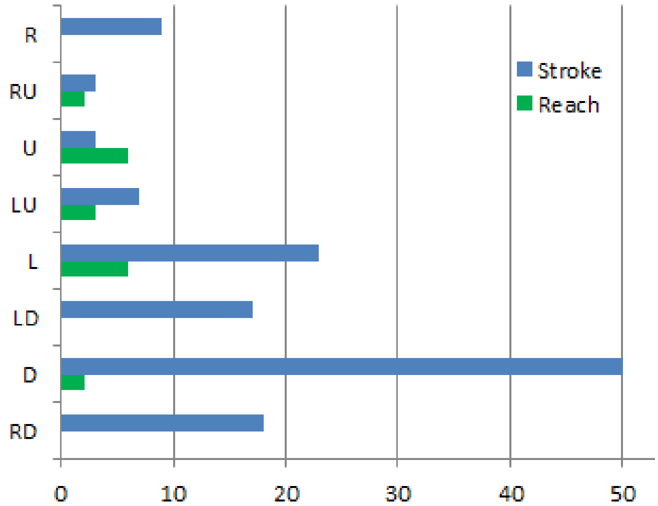


Figure 3-6: *Error number for error directions and selection techniques*

Further analysis using a two-tailed dependent T-test found that visual feedback had no significant effect on error rate with Stroke ($t_{127} = .49, p = .63$) but significantly reduced errors ($t_{127} = 2.30, p < .05$) with Reach.

We also noticed that when users make errors selecting a target direction, different directions have a difference chance of being selected by mistake. The number of Down directions selected in error was more than twice the number for the second most common error direction, Left, followed by Right-Down and Left-Down. While many fewer errors happened with the Reach selection technique, Up and Left attracted more errors than other directions and Right, Left-Down and Right-Down saw no errors. (Figure 3-6)

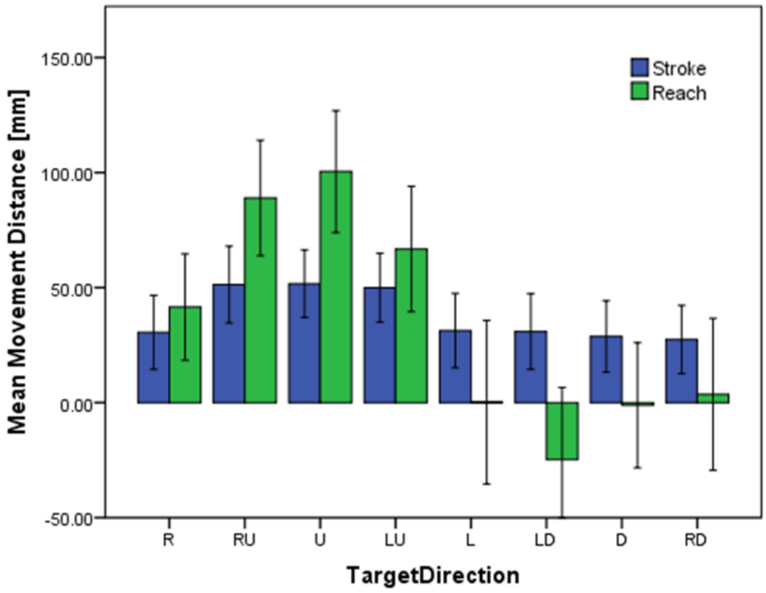


Figure 3-7: *Hand movement in the Z dimension when selecting in different target directions with the Stroke and Reach techniques.*

3.2.3.3 Hand Movement in 3D

The user's hand movement in 3D space was recorded during the selection tasks. Figure 3-7 shows mean hand movement distance perpendicular to the target direction plane (Z dimension) across all the trials by every participant. A minus value means the hand finishing position was behind the starting position (i.e. closer to the user's chest), and a positive value means the finishing position was in front of the starting position.

A repeated-measures ANOVA was used for Technique x TargetDirection. We found main effects for TargetDirection [3] ($F_{2.45,36.71} = 33.89, p < .001$), but no significant effect was found for Technique ($F_{1,15=}.17, p = .69$). We found an interaction effect for Technique x TargetDirection [3] ($F_{3.15,47.31} = 22.71, p < .001$).

3.2.3.4 User Preference

Figure 3-8 shows user preferences for Technique x Feedback and directions (from 1 for strongly dislike to 10 for strongly like). Participants preferred Reach because they felt Reach was accurate without sacrificing selection speed too much, so they felt confident selecting with Reach. Most participants liked the visual feedback as it is similar to the cursor in the desktop applications with which they are more familiar.

Right and Down were the most preferred target directions, while participants disliked Left-Down, Left-Up and Left. They commented that because they selected using their right hand, it was comfortable to move the hand to the right, and their hand was

[3] The sphericity assumption was not met so the Greenhouse-Geisser correction was applied; the corrected degrees of freedom are shown.

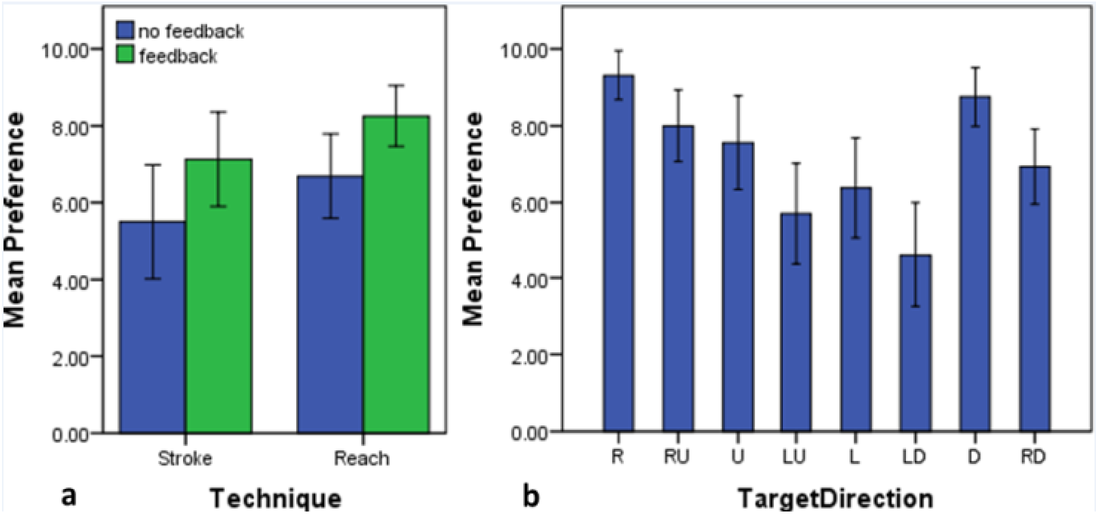


Figure 3-8: *User preferences for Technique x VisualFeedback and for TargetDirection*

more relaxed when moving downwards. In contrast, moving the hand upwards was more tiring, and they did not like Left-Down because they made more errors than in other directions.

3.2.4 Discussions

In this subsection, we provide some detailed discussions about the effect of different design factors and user behaviour based on the experimental results and feedback.

3.2.4.1 Effect of Selection Techniques

The results show that Reach is slower but more accurate than Stroke. However, if we look into the details, we find that Reach is a better candidate for freehand menu design. For example, if we compare Reach and Stroke with continuous visual feedback, we find the selection time with Reach (0.82s) is only 20.1% slower than Stroke (0.67s), but the errors with Reach (0.55%) are less than 10 times those with Stroke (6.88%). User feedback also favours Reach: users preferred Reach and commented that they could select with more confidence using Reach.

With the Stroke technique, we also found that some participants liked to move their hand a short distance in the opposite direction first and then perform the selection gesture. We speculate that this may have made selection more comfortable, but it caused erroneous selection in the opposite direction. For example, when the user tries to select Up, it is easy to move the hand too fast (i.e. faster than the speed threshold for Stroke selection) downwards under the influence of gravity (Figure 3-6).

Selection with Stroke took less time to Right and Down than to Up, suggesting that freehand gestural interaction using hand movement speed as the recognition parameter should consider this difference, especially when a single remote camera is used for tracking. Freehand gestural interactions based on hand position reach should minimize Left-Down movement using the right hand since it is error-prone and disliked by right-handed users.

3.2.4.2 Effect of Visual Feedback

Continuous visual feedback tends to slow down the selection time a little while improving the accuracy only of Reach. This is mainly because the Reach technique relies on hand location, similar to using a mouse in desktop applications. Selection became easier when the user could see the cursor movement. If feedback was not available, users had to estimate their hand position, and more errors occurred. Stroke, on the other hand, uses hand speed/direction rather than location so does not benefit from the visual feedback. This suggests that designers should consider providing a cursor for hand position based interaction, and perhaps not for gestures which do not rely on hand/body position.

Despite the increase in selection time, participants preferred the visual feedback. They said that the feedback gave them a better idea of their hand position and made selection easier. The Reach technique with feedback received the highest preference score, perhaps due to its accuracy and similarity to familiar desktop applications. The Stroke technique without feedback, which was not very accurate, had low preference ratings even though it had the fastest selection time. This also suggests that accurate and instant feedback is an important design element for freehand gestural interaction.

3.2.4.3 Effect of Target Direction

We found very strong directional effects in freehand menu selection. For example, users quickly generate speed downward, and right-handed users quickly reach a given distance rightward (Figure 3-3). This may be the result of both gravity and human arm movement patterns, i.e. more elbow movements were involved when the hand was moving upwards. Users need longer selection time for Left-Down with Reach, which may also be caused by the arm movement pattern: the right arm needed to go across the body, which involved more body movement and introduced more errors than the other directions with both techniques.

We also found that users made about twice as many errors when selecting off-axis directions than on-axis directions, with both techniques (Figure 3-5). As no such strong directional effect has been found with other devices such as touch screen [113, 191] or stylus [198]. This is probably because with freehand gestural selection users have to move their hands in the air without any support for a relatively long distance compared to using a touch screen, so the error introduced by the same angle is larger.

Thus designers should be careful when carrying over findings from other interaction techniques to freehand gestural interaction design.

3.2.4.4 Hand Movement

Because the directional gesture was performed immediately after the menu invoking pull gesture, the start position was near to the user's body and the user's hand tended to move forward while selecting the target. The results indicate that although the menu items were located in the same (vertical) plane, the corresponding hand movement towards the target did not stay in the same plane. And hand movement perpendicular to the user's chest varied with the technique and target direction. And in both techniques, the user's hand moved for a longer distance in the Up, Right-Up, Left-Up directions.

With the Stroke technique, hand finishing position for all target directions was in front of the starting position. This is possibly because Stroke is faster and the hand movement distance is shorter, so the hand end position is in the front of the start position. For the Reach technique, the hand movement distance in the Z dimension varied more across target directions, possibly because the hand moved a longer distance. In Up, Right-Up and Left-up, the user's hand still moved forward farther than with other target directions, while hands stayed close to the starting depth with Left, Down and Right-Down. One interesting exception was Left-Down with the Reach technique, in which the hand moved backward rather than forward. This is consistent with users' comments that they needed to rotate their arm and body to select the Down-Left target.

3.3 Option Selection in 3D Layout

In the first study, we focused in detail on option selection with different gestures and visual feedback in a 2D menu layout. However, as noted above, freehand tracking may be less accurate than using devices such as a touch screen or mouse, so potentially requiring larger menu items and limiting menu item size and number. Since one advantage of freehand interaction is that gestures can be tracked in 3D, menus could be displayed and selected in 3D space, which may help in presenting menus with more items without decreasing menu item size. Although considerable research has been conducted on menu selection techniques in 3D environments, most – like the first study above – have used 2D menus and there are no well established 3D menu selection techniques, especially for freehand gestural interaction. So we conducted a second study on the design and evaluation of option selection in a 3D menu layout [141].

3.3.1 Design

Again, as there is no button clicking or touchscreen tapping available with freehand interaction, marking menus provide a potentially fruitful approach to freehand menu design because selection can readily be performed with marking menus solely by gesture, without clicking or tapping. Many marking menu selections in previous research were based on 2D displays with 2D pointing input devices. In this work, we extend the use of marking menus to 3D environments with 3D freehand gestural input. This brings several challenges in 3D marking menu representation and 3D freehand gestural selection. For example, the 3D marking menu items must be rendered carefully to make sure the user can see all the menu items and perceive their 3D locations correctly in order to select them, and it can be challenging to select the target menu item from a 3D menu which is densely populated by menu items. Furthermore, as the menu items are distributed in three dimensions, hand movement in each direction (e.g. up, down, left, right, forward and backward) may be used to select corresponding menu items; which brings more challenges than moving on a 2D surface.

3.3.1.1 Gesture Design

Although ray-casting techniques have been shown to be effective for 3D selection, the origin of the ray is normally located in front of the objects. However, in a 3D marking menu, the ray origin is, by definition, located in the center of the objects, i.e. with the menu items arranged in three dimensions around it. Also, with ray selection using an input device in the user's hand, it is easy to confirm the selection by clicking a button, but this is not available in freehand gestural interaction. Furthermore, as the hand position is virtually surrounded by menu items in all three dimensions, there is no extra dimension and very limited space available for an additional confirmation gesture following the selection gesture. For these reasons, we designed our freehand gestural 3D menu selection technique based on the hand extension metaphor rather than ray-casting.

As in the previous study, we designed two menu selection techniques to enable selection and confirmation with a single directional gesture. In this study, we again began with the Stroke technique and the Reach technique, modified for 3D interaction. In cross selection [2], the target is selected simply by moving the cursor across the target's boundary in 2D. For interaction with a 3D menu, we designed the corresponding Reach technique so that the user selects the target by moving the cursor through a corresponding plane, as shown in Figure 3-9b.

Menus are usually invoked by a mouse click or pen touch, however, these inputs are not available in freehand gestural interaction, so we need to find a way to invoke the menu. Again, as the menu items may be distributed in all dimensions, movement of a single hand may not be suitable because it is easy to confuse the trigger gesture

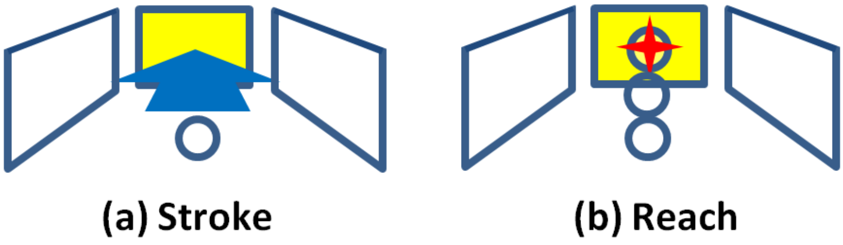


Figure 3-9: *Selecting an item using gesture with 3D menus (Top-Front view). (a) Stroke, using the movement's direction to select. (b) Reach, using the cursor position to select; the cursor must go through the target plane.*

with a selection gesture. One benefit of motion tracking without fiducial markers is that the tracking is not limited to the marked body parts; in contrast, it can track the movements of the whole body. This enables using movement of other body parts as the menu trigger gesture. And in contrast to a 2D menu, the menu items in 3D menu are distributed all directions in 3D space. This means that it is difficult to use the hand performing menu item selection to invoke the menu, as hand movement in any direction could trigger the menu selection incorrectly. Thus, we introduce the "hands up" gesture with the other hand, i.e. moving the hand up towards the shoulder. We use this gesture because "hands up" is very common in everyday life and so is easy to understand and perform, and can be easily separated from the gesture to select a menu item with the other hand.

3.3.1.2 Display Design

We chose the number of menu items for freehand selection based on previous work with 2D marking menus. With pie cursor [58], four and eight items were tested, and Escape [191] used 8 directions for its icon arrows. More items have been tried in [198] and results showed slower selection time and higher error rates for more than eight menu items. Zhao et al. [197] also conclude that to maintain acceptable accuracy rates, marking menus should not exceed 8 items.

Given that the movement ranges of a thumb [191], mouse [58], or handheld tracking device [68] are smaller than an arm, the radius of the menu and distance between each menu item could be larger with our freehand gestural selection. This makes our menu potentially able to handle more menu items.

Taking all these constraints into account, we constructed the 3D menu by putting 8 menu items in the vertical plane centred on the initial cursor position, 3 items in the front and another 3 behind the cursor position. These 6 items were in the same horizontal plane. So, we had 14 menu items in total.

Grossman and Balakrishnan [68] suggested that target size dimension along the main selection direction has a greater effect on performance than in the other two dimensions, however, this conclusion is based on the “move and click” selection technique using a 6-dof tracker equipped with a button. In our selection technique for freehand gestural selection (Figure 3-9), users only need to move their hand toward the target direction (with Stroke) or pass through the target plane (with Reach). Thus, all menu items in our 3D menu were 2D rectangles, so their size along the main selection direction approximated to zero.

Various types of presentation have been used for marking menus in 2D, such as pie [58], radius zone [73], rectangular zone and polygon [197]. We found polygon to be a good presentation to facilitate both Stroke and Reach. For a polygonal 2D marking menu, each menu item is a line occupying some length. If these items were selected by freehand gesture, we could consider the depth of the items as infinite. But for a 3D menu, since there are items in front and behind, the depth of the menu items in the horizontal plane cannot be infinite. Thus, rather than a line in a 2D plane, each such menu item is represented as a vertical rectangle at a given position in 3D space (see Figure 3-10).

In pilot trials, we found that if we render all the menu items as rectangles, their visual representations are similar and the user will have difficulty in perceiving the item’s location in the 3D environment. So we rendered only the 6 menu items in the front and back as rectangles, and rendered the other 8 menu items (in the vertical plane centred on the cursor position) as lines in order to provide visual separation; but the effective selection areas for all menu items were still rectangles.

Considering that the range of human hand reach both from Up to Down and from Right to Left is almost the same (about 2 arms’ length), we used an octagon as the layout of the menu items in the vertical plane centred on the cursor position, with each item having the same distance to the center. We could also apply the same layout for items in the front and back, giving the first layout design (Figure 3-10a, Octagon).

However, hand movement range forward and backward is shorter (about 1 arm’s length without body movement). To accommodate this difference, we also tested a rectangular layout for the items in front and back (Figure 3-10b, Rectangle).

Considering that the arm movements forward and backward are not symmetrical, e.g. there is more space when the arm is moving forward, while the hand is obstructed by the upper body when moving backward, we also evaluated non-symmetrical layouts, i.e. the combination of octagon in front and rectangle behind (Figure 3-10c, Octagon/Rectangle), and rectangle in front and octagon behind (Figure 3-10d, Rectangle/Octagon). For all four conditions, the 8 menu items in the perpendicular plane centred on the cursor were in the same locations. The 3D marking menu names reflect the layout of the menu items in the front and back.

In an initial study, we found that the Stroke technique produced a lot of errors in 3D menu selection. This is probably because the difference in direction between 14 menu

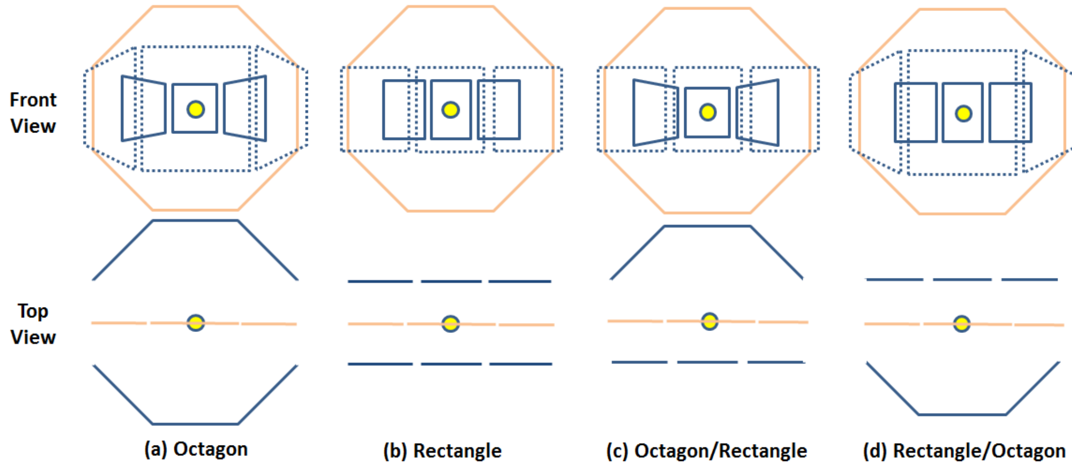


Figure 3-10: *3D menu display design. In the front view, the rectangles drawn in dotted blue lines represent the menu items at the back (i.e. closer to the reader), the rectangles in solid blue lines represent the items in front, and the orange lines represent the menu items located in the vertical plane around the initial cursor position. In the top view, the blue lines represent items in front and back, and the orange lines represent items located vertically. The yellow sphere in the center represents the initial cursor position.*

items is too slight for the hand to separate them effectively. So we carried forward only the Reach technique for investigation in the experimental evaluation reported in the following section.

Given that moving forwards and backwards to select targets is significantly slower than moving left and right [68], and the relatively short range of forward and backward hand movement, we tried to reduce the Z-depth of the 8 menu items located vertically, so that the 6 items in the front and back could be closer and therefore easier to reach. However, we found that selection errors for the 8 vertical items increase quickly when their depth becomes smaller than their length. So we maintained the selection area of these 8 items as square in the following study.

3.3.2 Experimental Evaluation

We conducted a controlled experimental evaluation to investigate the effect of different 3D layouts and target directions using freehand gestural selection. The experimental setting was the same as the previous study in Section 3.2.2.1 and is shown in Figure 3-11.

3.3.2.1 Independent Variables

The independent variables were 3D menu layout (Octagon, Rectangle, Octagon/Rectangle, Rectangle/Octagon, as shown in Figure 3-10) and TargetDirection (Up, Down, Left, Right, Front, Back, Left-Up, Right-Up, Left-Down, Right-Down, Left-Front, Right-Front, Left-Back, Right-Back).

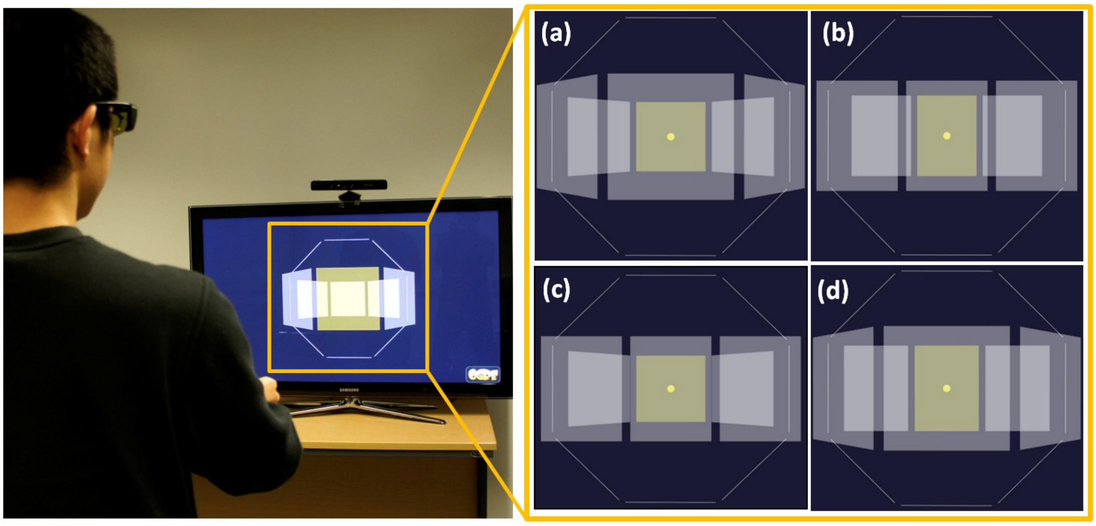


Figure 3-11: *Experimental setting and snapshots of screen. (a) Octagon (b) Rectangle (c) Octagon/Rectangle (d) Rectangle/Octagon.*

3.3.2.2 Hypotheses

- There will be significant differences in selection time and error rate when participants select with the different 3D menu layouts.
- There will be significant differences in selection time and error rate when participants select in different directions.

3.3.2.3 Participants

We recruited twelve volunteers (nine male and three female), between 22 and 30 years of age with a mean age of 25.58 (sd = 2.15). They were all right-handed and had some experience of gestural interaction such as gaming.

3.3.2.4 Procedure

Each trial started with a "hands up" gesture with the left hand. When the gesture was performed, a 3D menu appeared with the target menu item highlighted in yellow. The user's hand position was represented as a yellow sphere (diameter 5 cm in virtual 3D space) in the center of the menu. This cursor continuously followed the user's right hand movement.

The 8 vertical menu items were 80 cm from the center of the menu in virtual 3D space, and their width was 66 cm (the orange lines in Figure 3-10). Although they were rendered as lines for visual separation from other items in front and behind, their effective selection depth was 66 cm, so all the menu items located vertically were squares in motor space. In the different 3D menu layouts, the size and distance of the menu items located vertically were the same, while the size and distance to the center of horizontal items differed as a feature of the layout. In the Octagon menu layout (Figure 3-10a), the other 6 menu items located at the front and back were squares of the same size (66 cm x 66 cm). The menu in Rectangle layout (Figure 3-10b) had rectangles sized 66 cm x 53 cm. In the other hybrid layouts (Figure 3-10c, 3-10d), their size was the same as the corresponding menu item in the Octagon or Rectangle layout. All the menu items in the front were rendered as translucent in order to show the other, occluded, menu items.

User movement was mapped from real space to the visual 3D space as 0.3:1 (when the user's hand moved 3 mm in real space, the cursor moved 1 cm in virtual 3D space). A selected menu item disappeared when a selection was detected. If the user selected an item other than the target, an error sound was played. If the target was selected, the user was instructed by text on the display and an audible beep to start the next trial.

After the whole test was finished, the participant answered a questionnaire on menu layout and target direction preferences. They were asked to rate their preference for the different layouts and target directions from 1 (strongly dislike) to 10 (strongly like), and provide comments about their preferences. The whole study took about 45 minutes per participant.

3.3.2.5 Design

A repeated measures within-participants design was used. There were 4 sessions, one for each 3D menu layout. The order of the sessions was randomized. In each session, there was a practice block of 52 trials (3 trials in each direction), followed by another 2 test blocks with 52 trials each. Thus for each participant, 624 trials were performed in total.

3.3.3 Result

The selection time, error rate, and user preference were recorded and the collected data are analyzed in this subsection.

3.3.3.1 Selection Time

A repeated-measures ANOVA for 3D marking menu layout x TargetDirection was used to analyze the results. Main effects were found for 3D marking menu layout ($F_{3,33} = 5.00, p = .006$) and TargetDirection ($F_{13,143} = 7.47, p < .001$). An interaction effect was found for 3D marking menu layout x TargetDirection ($F_{39,419} = 2.58, p < .001$). Mean selection time for Octagon was 1.31 s, for Rectangle was 1.13 s, for Octagon/Rect/angle was 1.21 s, and for Rectangle/Octagon was 1.20 s.

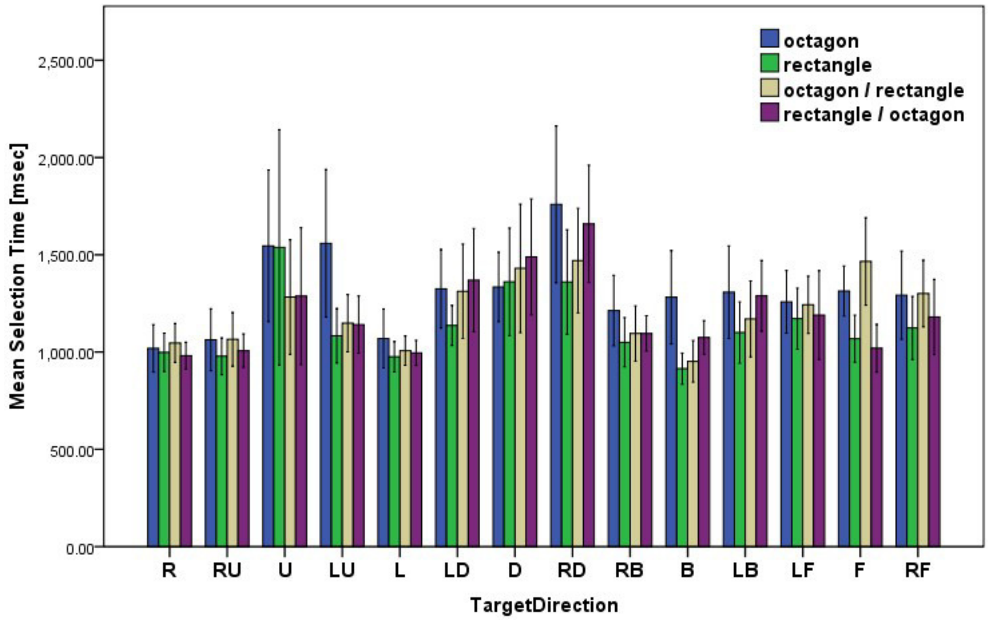


Figure 3-12: *Selection time for menu layout x TargetDirection.*

Post hoc Bonferroni pairwise comparisons showed that Rectangle was significantly faster than Octagon ($p = .022$). Mean selection times across the conditions are shown in Figure 3-12.

Further analyses using one-way repeated-measures ANOVA for TargetDirection in the four different menu layouts were performed. As illustrated in Figure 3-13, with Octagon each of Right and Right-Up was significantly faster than Front and Right-Down. With Rectangle, Left was significantly faster than Left-Down. With Octagon/Rectangle, each of Left and Back was significantly faster than Up, and with Rectangle/Octagon, each of Right, Left, Front, Right-Up, and Right-Front was significantly faster than Right-Down (($p < .05$) or more significant for all pairs). In Figure 3-13, directions represented by blue arrows are significantly faster than the directions in green arrows with each layout.

We also compared the selection time in the on-axis and off-axis directions. A two-tailed dependent T-test found no significant difference (t11=-1.51, p=.16).

3.3.3.2 Error rate

Users were required to select the target successfully, if necessary trying more than once. Each failure to select the correct target was recorded as an error. The first erroneous direction when users failed to select the target menu item was also recorded as "error direction" for this "target direction". A repeated-measures ANOVA was used for 3D menu layout x TargetDirection, as shown in Figure 3-14.

We found a main effect for TargetDirection ($F_{13,143} = 3.02, p < .001$), but no signif-

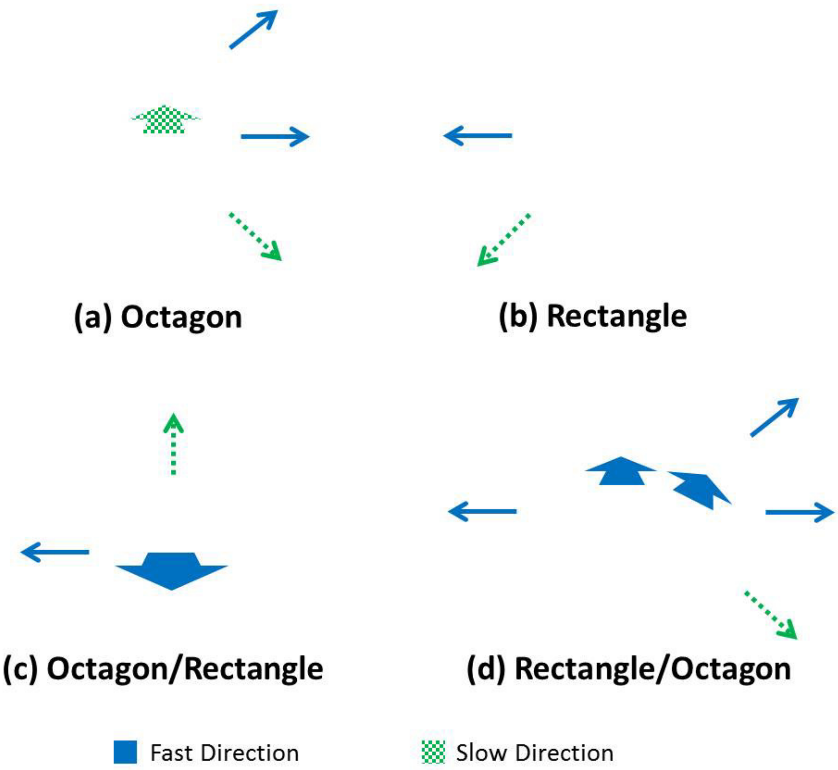


Figure 3-13: *Selection time comparison in different directions. Blue directions took significantly less time than green directions (p¡.05). Line arrows represent directions in the vertical plane, and block arrows represent directions in the horizontal plane.*

icant effect was found for menu layout ($F_{3,33} = 0.77, p = .52$). We also found an interaction effect for 3D menu layout x TargetDirection ($F_{39}, 429 = 2.12, p < .001$). Mean error rate for Octagon was 5.26%, for Rectangle was 6.94%, for Octagon/Rectangle was 5.55%, and for Rectangle/Octagon was 5.06%.

We also analysed errors in on-axis directions, i.e. Right, Down, Up, Left, Front, Back, and off-axis directions (Figure 3-15). A two-tailed dependent T-test found that whether the direction was on-axis or off-axis had no significant effect on error rate ($t_{11} = -2.04, p = .07$).

The distribution of error directions is illustrated in Figure 3-16. The top 4 mistakes in each 3D menu layout are shown. The user's hand moved to Back very easily when trying to select Right-Back (a, b, c), and to Right-Back easily when the target was to the Right and the Octagon layout was used at the back (a, d). In the second and third rows of (b, d), we find that the user's hand moved to Front easily when selecting Up, and moved to Left-Front easily when selecting Left-Up if the Rectangle layout was used at the front. The user also moved her hand easily to Left when trying to select Left-Back and Left-Front in the Octagon layout (a, third and fourth rows). Overall, forward and backward directions produced most of the errors shown in Figure 3-16,

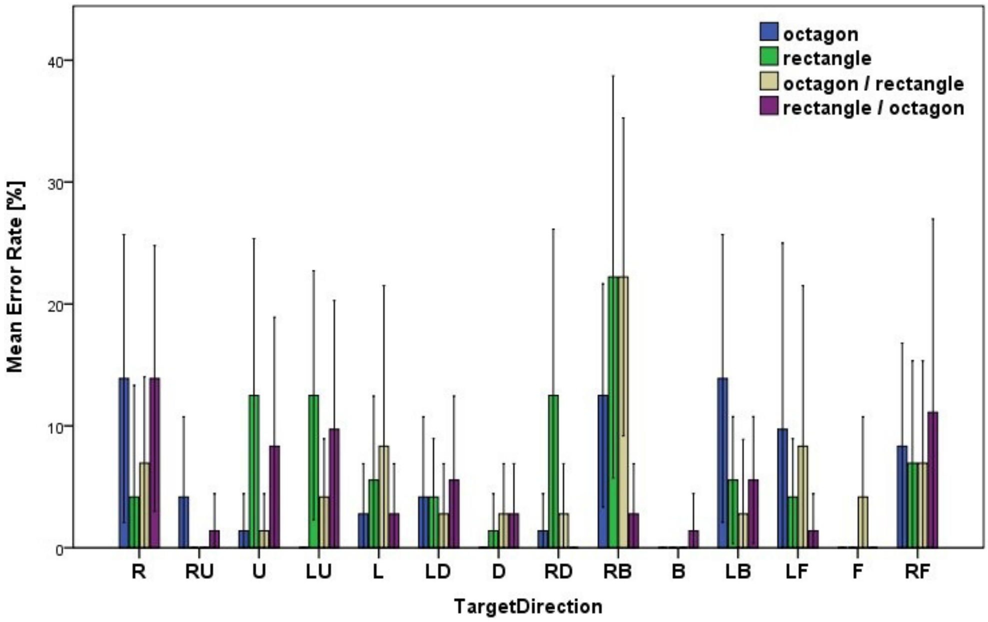


Figure 3-14: *Error rates for 3D menu layout x TargetDirection.*

backward producing more than forward.

3.3.3.3 User Preference

User preference data on the 3D menu layouts and target directions were collected by questionnaire. Figure 3-17 shows user preferences for the 3D menu layouts. A one-way repeated-measures ANOVA for menu layout found no effect ($F_{3,33} = .33, p = .801$). Although there is no generally preferred 3D menu layout for all participants, some participants showed strong personal preference for certain layouts. With respect to visual feedback, some participants preferred the combinations of Octagon and Rectangle because they provided visual differences between the items in front and behind and facilitated their identification of item position. However, some other participants preferred the Octagon layout because it represented the depth difference of items in front and behind more clearly, and the symmetrical shape made them feel comfortable. The visual feedback in the Rectangle layout was disliked by users because the items in front and behind looked similar, however, some participants preferred Rectangle because the items in front and behind are all in the same plane, so it was easier for them to locate and select the items by right angled hand movements, e.g. moving the hand a little to the right and then forward to select Right-Front, rather than making a diagonal movement.

Some participants also commented that they preferred the Octagon layout when it appeared in the back, because it is not easy to control the hand moving backward, and the Octagon layout gave each item more space, making it easier to select. They even

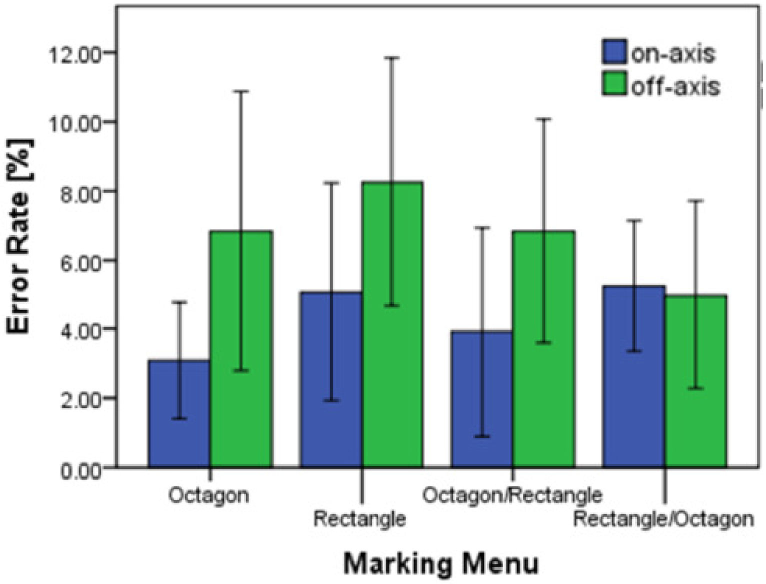


Figure 3-15: *Error rates for on-axis and off-axis TargetDirection and 3D menu layout.*

suggested putting only one item at the back and removing the Left-Back and Right-Back options. This comment aligns with the error analysis in Figure 3-16. Participants felt that selecting forwards was more accurate than backwards, although it required a little more effort.

Figure 3-18 shows user preferences for (a) all TargetDirections and (b) on-axis and off-axis TargetDirections. A one-way repeated-measures ANOVA found a main effect for TargetDirection ($F_{13,143} = 4.18, p < .001$); Right and Front were the preferred target directions, while participants disliked Left-Down and Left-Back. They commented that because they selected using their right hand, it was comfortable to move the hand to the right. It was also comfortable moving the hand forward. In contrast, when moving the hand to Left-Down and Left-Back, the hand had to move across the body close to the chest, making them feel uncomfortable.

A two-tailed dependent T-test found significantly higher preference for on-axis directions than for off-axis directions ($t_{11} = 4.77, p < .001$), as shown in Figure 3-18(b). Users commented that when they saw an on-axis target, they perceived the direction and started to move the hand in that direction easily and confidently, while for off-axis directions they needed to hesitate to think about the direction and moved the hand more carefully. In other words, the off-axis directions probably required more mental and physical workload than the on-axis directions in our 3D marking menus.

Users also made some other suggestions about the visual feedback of the cursor. They said it was not easy to select if they did not know the relative location of the cursor and the 3D menu items, and the stereo 3D display did not provide enough clues about this relative location. So it would be better if the cursor projections on each

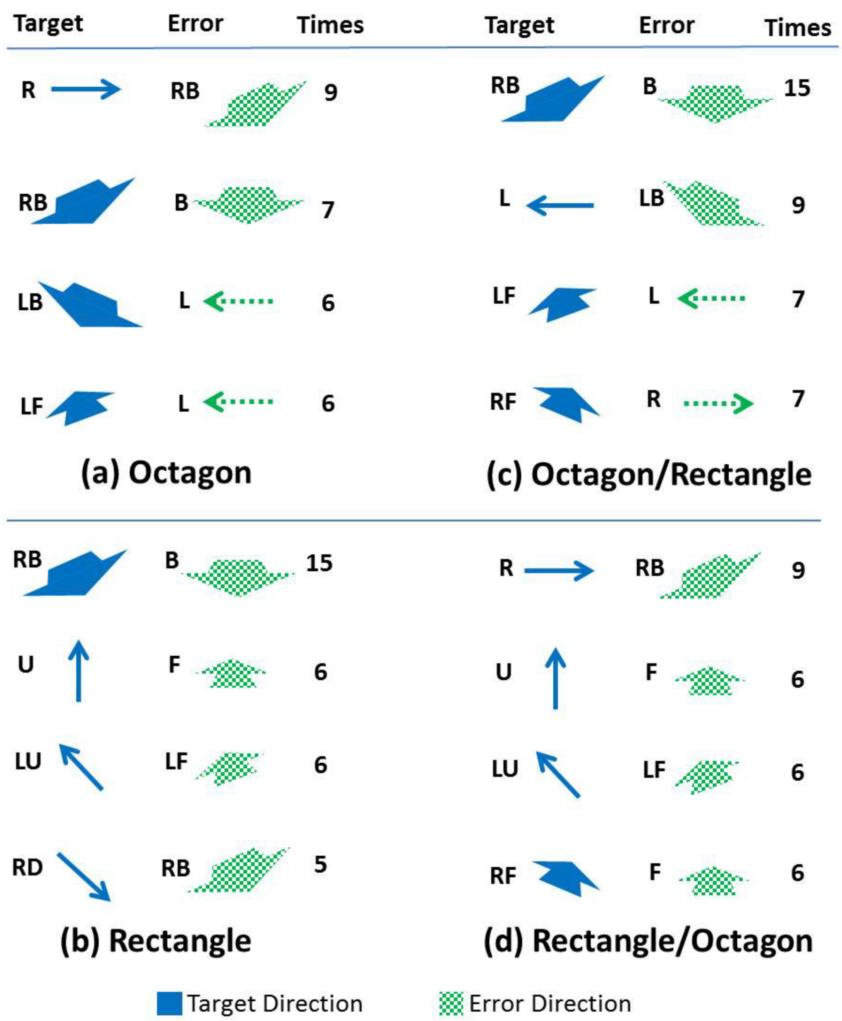


Figure 3-16: *The first direction selected by mistake (green arrows) when the user tried to select the target direction (blue arrows). Line arrows represent directions in the vertical plane, and block arrows represent directions in the horizontal plane.*

dimension could be displayed on related menu items, so the user has a clear perception of the current cursor position relative to the menu items in the 3D environment.

3.3.4 Discussions

Based on the experimental results, the effects of different design features and a comparison with previous designs are discussed in this subsection.

3.3.4.1 Effect of Layout

The Rectangle layout was significantly faster than the Octagon layout, while there were no significant differences in error rates between them. The different layouts of Rectangle and Octagon make the Rectangle menu items located in the horizontal plane smaller in

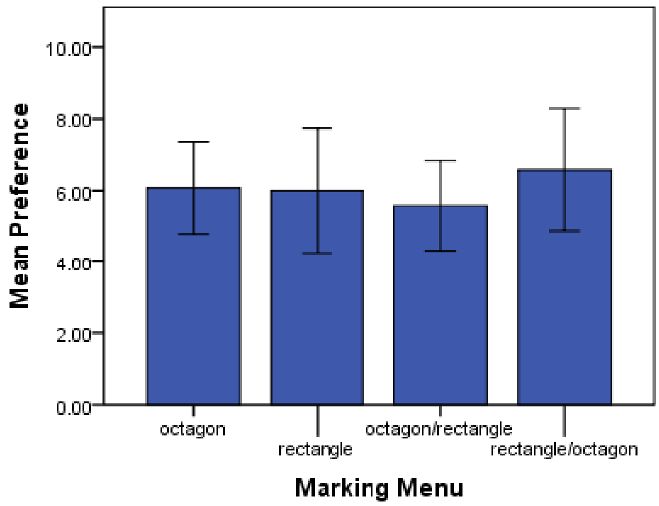


Figure 3-17: *User preference of different layout.*

size but closer to the menu center than with Octagon (Figure 3-10). The results suggest that these differences made the Rectangle layout faster than Octagon using freehand gesture without significantly sacrificing accuracy. However, although the error rate with Rectangle was not significantly different, we found that the smaller space of the Rectangle layout tended to produce more errors in the forward and backward directions (Figure 3-16b, 3-16c, 3-16d). Users also noted that the menu items in front and behind with Rectangle looked similar, affecting the perception of their 3D location. These findings suggest that Rectangle could still be improved. For example, we can reduce the number of menu items behind the hand location and increase their size. This could potentially not only introduce more visual difference, but also reduce the error rate when selecting items behind the hand location.

3.3.4.2 Effect of Target Direction

Significant effects of target direction were found on both selection time and error rate. Directions towards Right and Front were comfortable for our right-handed participants and on-axis directions such as Up, Down and Back were also preferred; users did not like Left-Down and Left-Back. So, in designing 3D menus for gestural interaction, we could consider using only on-axis directions if higher accuracy or fewer options are required. When there are many options, we could put high priority or commonly selected options in the on-axis directions.

We also found that users had difficulty selecting the targets located vertically, especially downwards (Down, Left-Down, Right-Down). This may be caused by the arm movement pattern: the hand usually does not move in the same vertical plane as the user's chest. For example, when the hand moves to Right-Down, it also moves back at the same time. Although people can try to adjust their action for moving their hands

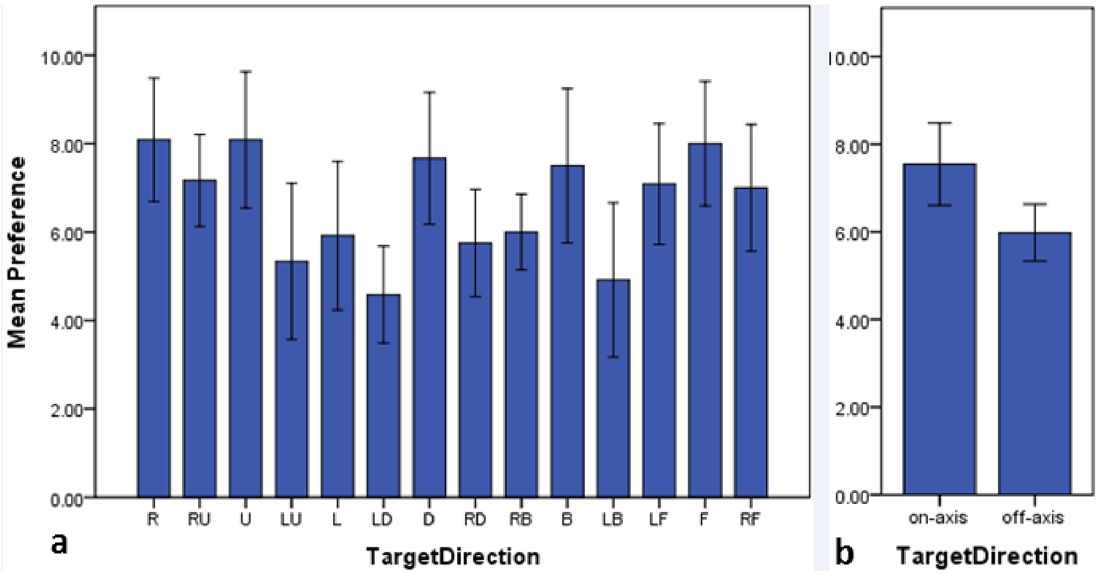


Figure 3-18: *User preferences for (a) all TargetDirections and for (b) on-axis and off-axis TargetDirections.*

in a vertical plane, such as involving more elbow movement, it increases the physical demands of the arm movements and requires users to pay more attention to their hand location. This suggests that the layout of items in the vertical plane could also be improved.

3.4 Design Space Comparison

The set of 2 related studies reported here contributes to our knowledge of designing for 3D freehand interaction, leveraging and extending knowledge from previous experiences in the design of gestural interaction with 2D touchscreen surfaces and with 3D interaction enabled by hand held devices. In this section we provide some further discussion of the interaction design features, user performance and behaviour, comparing 3D freehand gestural interfaces with 2D touchscreen gestural interfaces and 3D gestural interfaces using hand-held devices. We then introduce some further suggestions for freehand 3D interaction design.

3.4.1 Comparison with 2D Interaction Techniques

3.4.1.1 Similarities

Kurtenbach and Buxton [104] suggested that direct manipulation can be combined together with menus in the same 2D interaction, complementing each other effectively. And this also appears to be true with 3D freehand interaction. For example, with menu cone, users directly control the cone to cast the targets and use a marking menu to disambiguate the selection. Some general findings from 2D interaction may also apply in freehand interaction. For example, 2D marking menus have better performance when the number of menu items is even (e.g. 4, 8 or 12)[104] as these numbers benefit from a salient metaphor with circular layout (e.g. compass or clock).

The effect of visual feedback also has some similarities with 2D touch screen interaction and 3D freehand interaction. For example, investigations of typing with a 2D touch screen [78] shows that visualizing the touched positions using a simple dot decreases the error rate but also decreases the speed, which is quite similar to the visual feedback effect reported in the first study here.

3.4.1.2 Differences

When the gestural movements are constrained by a 2D surface, users can readily make approximately straight marks [107, 113]; thus it is easy to distinguish the selection gesture from other possible gestures [107]. In contrast, although the X-Y coordinates of freehand movements are also straight as on a 2D surface, the component of the movement in Z makes the entire path of the freehand movement curved in the 3D space, which may lead to some ambiguities and errors as observed in the second study.

3.4.2 Comparison with 3D Interaction using Hand-held Devices

3.4.2.1 Similarities

Compared to other marking menu selection techniques, such as using fingers on a touch screen [113, 191], we found freehand gestural selection enabled by a single remote camera had both similar and contrasting characteristics regarding target directions. For example, in Escape users disliked Left-Up, Left and Left-Down [191], similar to our results. However, no significant effect of target direction was found for time or error rate in Escape [191]. In [113], in contrast to our results, the stroke angular error on a multi-touch surface was small towards Down while big towards Right-up.

Grossman and Balakrishnan noted that moving the hand forwards and backwards to select targets was significantly slower than moving in other directions [68] with hand-held devices. And we have found similar results here for freehand 3D selection. In the Octagon layout in the second study, in which all targets are the same distance from center, we found selection forward to be significantly slower than right and right-up directions, which is similar to previous findings. The reason is likely to be similar to that suggested by Grossman and Balakrishnan: selection with smaller muscle groups can result in better performance than using larger muscle groups [108, 35].

3.4.2.2 Differences

Comparing our 3D menu layout to that in [67], in which the 3D menu is organized as a cube and the layout of the central plane is similar to the layout of Rectangle in this study, the significant effect of target position on error rate is found in both studies. However, the accuracy in different directions varies. Certain positions perform similarly, for example front and back are accurate in both studies, while some other positions have different accuracy, for example Down is accurate here, while error-prone in [67].

3.4.3 Comparison between 2D and 3D Studies

3.4.3.1 Selection Performance

With freehand gestures or hand-held devices moving in the air, the 3D movement input could be used in either 2D or 3D interaction tasks. Similar tasks were studied by Teather and Stuerzlinger [165] with a mouse or a 3D tracker. They tested the performance of the 3D input device in different settings, including using it in fully 3D space and using the 3D device to emulate a 2D mouse with and without physical 2D surface support. The results indicated that operation was faster and more accurate when the tracker was used in 2D rather than 3D mode.

We find a similar effect with freehand selection. Comparing the results from Studies 2 and 3, we find that selecting a 2D target with freehand gestures can be much more accurate than a 3D target. Even allowing that there are more targets in the 3D menu (14) than in the 2D menu (8), the difference in error rate is still much higher with about 10 times the error rate for 3D selection (more than 5% for each layout, using the Reach technique with visual feedback) than for 2D (0.55% with the Reach technique and visual feedback). And the selection time in 3D is also slower than in 2D.

The findings of 2D selection outperforming 3D selection with both freehand gestures and hand-held tracking devices could all be caused by the freedom of hand movement. For the 2D menu selection in the first study, only the X-Y coordinates of the hand movement are used in the selection, thus item depth in the Z dimension can be treated as infinite. However, in the 3D menu selection, all the X-Y-Z coordinates of hand movement are used, and the depth of menu items located vertically is much shorter (i.e. the same size as their length). As the user's hand tends to deviate in the Z dimension when moving sideways (Figure 3-7), it may lead to more errors (Figure 3-16).

3.4.3.2 Target Direction

Direction effects have been found in previous 3D selection research. For example, a selection task with the hand moving in 3D to reach a target in a fixed physical vertical board was evaluated in [124]. Their results show a significant effect of target direction on selection time, similar to our findings in Studies 2 and 3.

We compared the detailed results in [124] with out first study because they both involve moving the hand in 3D space to reach a target located in a 2D plane, and both have 8 targets layed out vertically. We found that the trend of selection time in different directions is very similar to the freehand Stroke technique in first study (e.g. selecting upwards is slower than other directions), rather than the freehand Reach technique. This is a very interesting finding considering that the selection technique design in [124] is much more similar to the Reach technique in every respect.

However, if we examine the details of the user behavior in 3D space, we find the similarity between the freehand Stoke technique and the selection task in [124]. In their experiment, the targets were on a fixed board located in front of the hand's starting location, so users always had to move the hand forward to select. From an analysis of the user's hand movement in 3D space in our study (Figure 3-7), we see that with the Stroke technique, users' hands always moved forward regardless of which direction they were actually selecting. Thus, the Stroke technique shares more similarity in user hand movement behaviours with [124], and it is therefore less surprising that Stroke has more similar selection times than Reach when compared to the experiment in [124].

This finding again illustrates the tendency for deviation in the Z dimension with freehand 3D gestural selection, and reminds us again that when designing techniques for freehand interaction, similar designs may produce very different user behaviours and performance.

3.4.3.3 Visual Feedback

We tested the Reach selection technique with and without visual feedback of the cursor in our study of 2D selection. Compared to another similar study in 3D menu selection [67], the common finding is that visual feedback improved the accuracy of the selection. However, the improvement of selection time with visual feedback in their study did not occur in our 2D menu study. This is possibly because there were more menu items and the layout was more complex in their study, so the visual feedback may have helped more, while our 2D menu selection here was straightforward. Hence, in the second study of 3D selection, we used the Reach technique with visual feedback because visual feedback can be more important for such complex tasks with more targets and a more complex 3D space layout.

3.5 Design Suggestions

Overall, the differences from previous research findings are largely due to differences in user behaviour. Moving the bare hand and arm freely in the air is different from holding and moving a mouse on the desktop, tapping the thumb on a mobile phone, or multi-touch movements with the fingers on a touch sensitive surface. The natural behaviour of the human body is different with different biomechanical parts (e.g. finger, hand, arm, shoulder), different ranges (e.g. small, medium or large area), and dimensions (e.g. 2D physical surface, 3D free movement in the air). Successful 3D freehand gestural interaction design thus depends in large part on understanding the user's perceptions, capabilities and behaviours, and bringing this understanding to bear on the design process. Designers should consider carefully whether or not findings from other gestural interaction techniques that involve handheld devices can directly be carried over to the design of freehand 3D gestural interaction.

Despite the challenges, 3D freehand interaction still has its advantages, such as hand movement in the third dimension increasing the scope for interaction design. Users can also leverage more of their experiences in real life with freehand interaction compared to established interfaces. Combining the results from the 2 studies reported here, some overall design guidelines emerge for 3D freehand interaction designers.

- The Reach technique, building on the goal crossing technique [2] in 2D interfaces, could be a useful method for target selection with 3D freehand gesture, as it does not require any confirmation trigger for selection. Although finger or hand poses could be used as a trigger, more precise tracking equipment would be required (e.g. data gloves [126]), or the tracking could be sensitive to viewpoint changes [157, 71] when using a single camera. Maintaining certain finger/hand poses may also introduce fatigue issues. With the Reach technique, on the other hand, users can select the target using their experience of reaching for an object in real life, without the need for a finger or hand pose as a gestural confirmation or delimiter.

- Even when working in 3D, designers should consider mapping 3D hand movements to 2D interaction for simple user interfaces with few elements. The number of targets located forwards or backward should be limited as the hand moves more slowly forward and backwards [68], and can be error-prone, as shown in the second study. Designers should also think carefully before committing to a full 3D interaction task, because such a task can be more demanding than a similar task in 2D and could be slower and more error-prone for freehand 3D interaction.

- One way to use hand motion in the Z dimension is as the delimiter or trigger to invoke certain commands or to make confirmations. For example, the pull gesture is used in the menu cone to connect between gestures (Figure 4-2). Given that a finger cannot currently be tracked reliably as a gesture delimiter by a single low-cost camera, and hand movements in the X-Y dimensions can be used for faster and more accurate actions, using hand movements in the Z dimension as the gesture delimiter is a potentially useful addition to the designer's toolkit.

- In both 2D and 3D selection, we found that users have preferences for different hand movement directions, e.g. they do not like cross-body movements such as left-down or left-back. Considering previous research on the hand's comfortable range for reaching static objects [97, 187] or dynamic objects [22], keeping the user's hand movements within the range of their comfort zone, and avoiding uncomfortable areas or directions, is important for freehand gestural interaction design.

3.6 Summary

In this chapter, we reported two studies about the option selection to address the challenge of gesture delimiters for freehand gestural interaction. The first study focused on freehand menu selection in a 2D layout, based on the previous research about 2D desktop and touch interaction design, we designed the selection techniques for freehand interaction. The user evaluation suggests the reach selection technique could be promising for freehand option selection. The second study extends the freehand reach selection technique to 3D space, and investigated the effect of different 3D layout and the directions in 3D. In both studies we explored interaction design, user performance, behaviours and preferences for freehand 3D selection techniques. After reporting the design and findings, we also present the guidelines for freehand interaction design. The results suggest that while design of freehand gestural interaction can leverage findings from other similar interaction techniques using more traditional input methods, designers should consider carefully whether they do in fact carry over. The work in this chapter laid a foundation for investigations in the following chapter about freehand gestural interaction in dense 3D environments.

CHAPTER 4
FREEHAND GESTURAL SELECTION IN DENSE ENVIRONMENTS

4.1 Introduction

In this chapter, we extended the work in the previous chapter about simple option selection to more complicated tasks such as object selection in dense 3D environments and text entry with virtual keyboard. As we analysed in Chapter 2, another main challenge for freehand gestural design is performing accurate operations, as the freehand tracking is normally noisy and the hand is moving in the air without any physical support. Our aim in this chapter is to analyse the requirements and design the effective freehand gestural interaction techniques for high accuracy tasks.

Building on the design and findings in Chapter 3 about the option selection design, we report two studies in this chapter focusing on freehand gestural interaction in dense environments. The first study is about target selection task in densely populated 3D space. And different from option selection, the targets have random locations in 3D space, and could be occluded by each other as well. Thus this selection task could be much more difficult than selecting an option from a menu as shown in Chapter 3. We reported design of the selection techniques, the user evaluations and the results. We found that interaction design requiring a high accuracy single action are not appropriate for freehand gestural selection, while separating it into several connected low demand operations could be a potential solution.

Then the second study in this chapter is about freehand gestural text entry with virtual keyboards, which is essentially a series of characters selection from a densely tiled keyboard. We presented 3 different input selection methods and 2 different keyboard design, and compared users' actual performance, behaviour, task load and preference in a 5 days period. We found that our proposed reach and expand selection method, which builds on the reach method in Chapter 3, could be a useful method for accurate target selection, especially used with circle layout.

In the end of this chapter, a series of practical design suggestions are provided based on the results and observations in the evaluations. We also presented design strategies for freehand selection tasks, as well as 5 design examples for selection tasks with different dimensionality and density.

4.2 Object Selection in Densely Populated 3D Environments

Although selection is a fundamental interaction task in most applications and considerable research has been conducted on selection techniques for 3D environments, most existing methods cannot be used directly in freehand interaction. The task is even more complex if the target is located in a dense 3D environment containing multiple objects, requiring a high precision interaction technique. In this section, we report the design and evaluation of two 3D freehand selection techniques, and discuss the implications of our findings for freehand 3D user interface design.

4.2.1 Freehand Selection Technique Design

In this subsection, we set out to design gestural selection techniques that are purely freehand.

4.2.1.1 Selection Metaphor

As ray selection has been shown to be effective in dense 3D environments [69, 168], our designs for a freehand selection technique drew heavily on ray selection. However, ray casting techniques normally need a high level of accuracy because only the object intersected by the ray can be selected. Since freehand motion captured by a low-cost camera can be noisy and jittery, it can be difficult to intersect the target object with the ray, especially in a dense 3D environment. Compared to ray selection, the cone cast selection technique [115] extends the ray to a cone shaped volume that has a more generous selection volume and can therefore make selection easier. So we take the cone selection technique as the basis for our freehand selection technique (Figure 4-1b, Figure 4-2a).

To use hand motion to control the cone direction, we fix the position of the cone apex and the position of the cone base center in two parallel vertical planes. The X-Y coordinate of the user's hand can then be used to control the X-Y coordinate of the cone base center. Thus, the user can control the cone direction with hand motion in the vertical X-Y plane, and the user's hand motion in the Z dimension can be used for a disambiguation or selection confirmation action. See section 4.2.1.3 and 4.2.2.5 for details of gesture design and implementation.

4.2.1.2 Disambiguation Method

For a ray selection technique in a densely populated 3D environment, the user typically first moves the ray to the target to select. If multiple objects are in the selection range, the user must disambiguate the selection with another selection.

Two common disambiguation methods are disambiguation using a marker on the ray (depth ray, lock ray), in which the user moves her hand to control ray direction and marker depth to locate the target; and disambiguation using a menu popping out from the 3D environment (flower ray, SQUAD). These disambiguation methods have been shown to be effective with handheld devices in densely populated environments [69, 98, 168].

Other disambiguation methods, such as using two rays to select, have relatively poor performance [69] or require movement and confirmation with both hands [186] so are less appropriate for freehand selection. It is also possible to disambiguate automatically with algorithms, but this method has been shown to be inconsistent in different 3D environments [150]. So we focused on the marker and menu disambiguation techniques for freehand selection.

4.2.1.3 Gesture Design

We designed two techniques for freehand gestural selection based on the cone cast technique, as well as marker and menu disambiguation techniques.

4.2.1.3.1 Marker Cone With marker disambiguation methods, the disambiguation may happen together with selection in a single phase, with only one "select and confirm" action, or in a separate disambiguation phase with two actions for first selection and then disambiguation. Given that the first approach (depth ray) has been shown to outperform the second (lock ray) [69], and it also requires only one confirm action, we designed our "marker cone" technique with selection and disambiguation together in the same phase.

With marker cone, the user controls the cone direction to contain the target inside, and disambiguates the selection by moving a marker forward and backward along the cone center. The closest object in the cone to the marker is selected. The direction is controlled by moving the hand in a vertical plane, and the marker is controlled by moving the hand perpendicular to this 2D plane. With ray selection techniques [69], only the object on the ray can be selected and disambiguated, which sometimes is not the closest object to the marker in 3D (Figure 4-1a). Leveraging the cone cast technique, the marker cone technique can select the closest object to the marker in

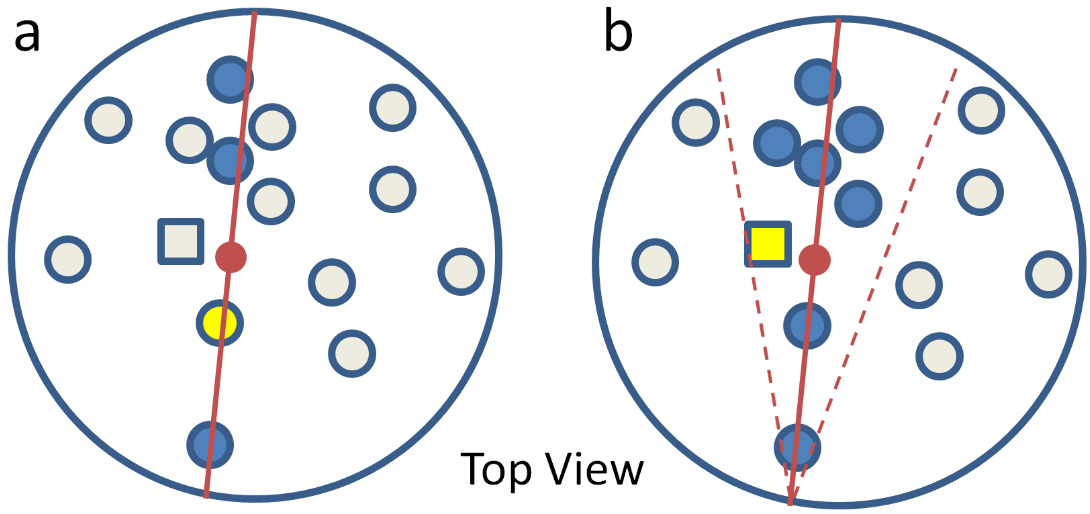


Figure 4-1: *The target is the cube. (a) Ray techniques select the object (yellow) intersected by the ray and closest to the depth marker. (b) In marker cone, the selection area is enlarged and the object in the cone closest to the marker is selected.*

3D (Figure 4-1b), giving it also the benefits of another selection technique for dense 3D environments called 3D bubble cursor [168], which can dynamically resize to select the closest target in 3D. Thus, the user can select the target without putting the ray precisely on the target, making the selection less demanding and easier to achieve with noisy motion tracking.

The marker cone requires the user to move her hand continuously in 3D space to select the target, which means it is difficult to make the confirm gesture using the same hand. Any movement with this hand will change the marker position and selection. One advantage of freehand interaction is that the motion of both hands can be tracked. So to address this challenge, we introduce a "hands up" gesture performed with the other hand as confirmation. When the target is selected, the user can confirm the selection by raising her other hand. Thus, the user can control the marker cone with one hand without needing an extra movement of that hand as a confirmation gesture.

4.2.1.3.2 Menu Cone With the menu disambiguation method, after the selection confirmation a menu pops out with the selected objects as menu items, and the user disambiguates the target by a menu selection. As two selections are required, so two selection confirmation gestures are needed. The menu used in the disambiguation phase is normally designed as a marking menu [69, 98] for fast selection. As marking menus also have the potential to enable selection with just a directional action rather than a selection action followed by a confirmation action, we used a marking menu in the menu cone technique for freehand selection.

To confirm the first cone selection, an extra action, e.g. button click [69, 168], is typically used with hand held devices, however, as no handheld device is available in

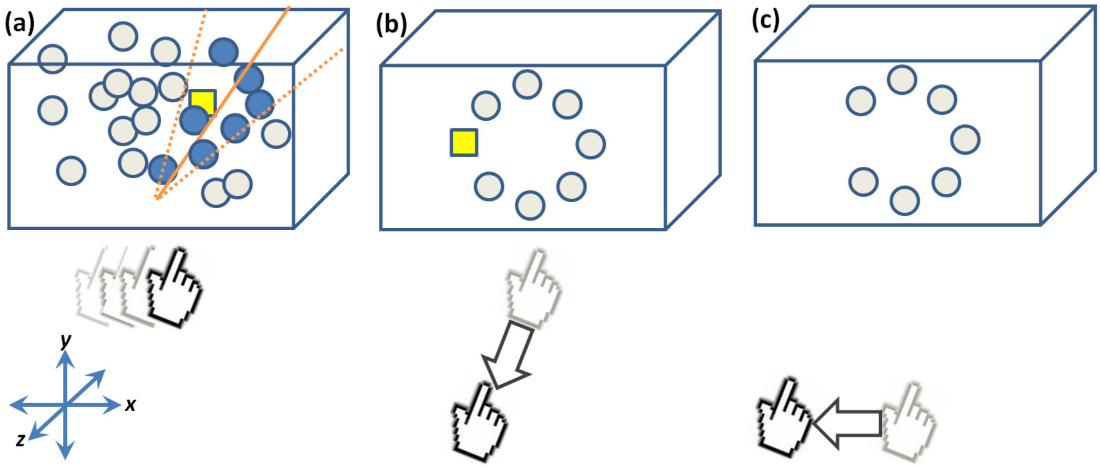


Figure 4-2: *Menu cone. (a) Point the cone to the target. The target turns yellow and distractor turns blue when selected. (b) Pull to confirm the selection and the disambiguation menu appears. (c) Perform a directional gesture towards left to select the target. The target disappears after selection. The definition of X-Y-Z dimension is shown in the left bottom, and applies to all following studies and discussions*

freehand gestural interaction, we cannot directly carry over these selection techniques. Similar to marker cone, the menu cone direction is controlled by moving the hand in an X-Y vertical plane. Thus, we use a quick hand movement perpendicular to this 2D plane along the Z dimension to confirm the first selection, as a “pull” or “push”. We chose “pull” to confirm the initial selection because it aligns with the visual perception of the marking menu popping out towards the user. Users can cancel and go back with a “push” gesture in the reverse direction if they make a wrong selection.

To confirm selection in a marking menu, a button click [69, 168] or finger touch/lift [113, 191] can be used to point to the menu item or specify the end point of the directional selection gesture. But again these methods cannot be used in freehand interaction, and we have to use hand motion. To determine the direction of the marking menu selection gesture, we use hand motion speed as the trigger: the user moves her hand in a direction, and the directional gesture end position is determined when the hand movement speed is faster than a threshold. The direction is then calculated using this end position and the hand position 0.1s earlier. With this method, the marking menu selection and confirmation can be performed by a small, quick movement in the direction of the target, as shown in Figure 4-2.

Since performing a confirm action with freehand is not as easy as with handheld devices, we use only one disambiguation, rather than repeated disambiguation as may be used with a handheld device [98]. Zhao et al. [198] showed slower selection time and higher error rates for more than eight menu items, therefore, we used 8 as the number of items in the disambiguation marking menu. The user first selects 8 objects with the cone, and then disambiguates the target from the other items in the marking menu.

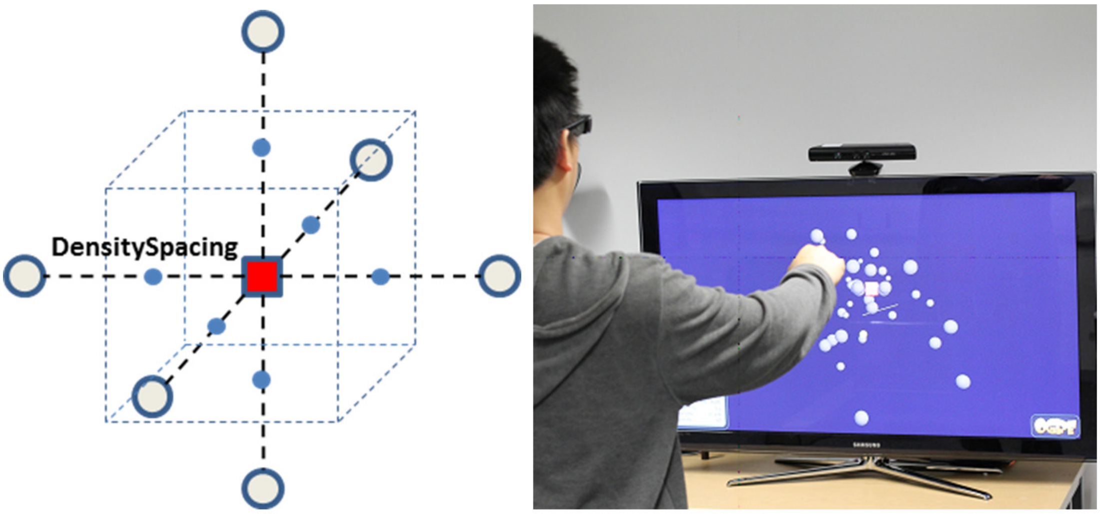


Figure 4-3: *Left: Six distractors were placed close to the target [168]. Right: Equipment and settings.*

4.2.2 Experimental Evaluation

We conducted an experiment to evaluate the freehand selection techniques, as well as the effect of 3D environment features such as density of objects, target visibility and the target distance.

4.2.2.1 Equipment and Setting

The 3D environment was displayed on a Samsung PS50C680 50" 3D plasma TV with 120 Hz refresh rate at 1280x720 resolution. Samsung SSG-2100AB/XL 3D active glasses were used by participants to view the display in stereoscopic 3D. The height from the ground to the center of the display was 107 cm, and the user stood 200 cm in front of the monitor. The user's movements were tracked using a Microsoft Kinect camera, with a refresh rate of 30 fps, and the OpenNI API on Windows 7. The Kinect camera was placed on top of the display, as shown in Figure 4-3.

4.2.2.2 Independent Variables

The independent variables were Technique (marker cone or menu cone), TargetDistance (100 or 200 cm from the start position to the target), DensitySpacing (30, 45 or 60 cm), and Visibility (Visible or Occluded). The start object position was the same in all trials and users selected it to start each trial. DensitySpacing is the distance from the distractors to the target, as in Figure 4-3. In the Visible condition, there were no distractors between user and target so the target was fully visible to the user. In the Occluded condition, two distractors fully occluded the target.

4.2.2.3 Hypotheses

- There will be significant differences in selection time and error rate when participants select with marker cone vs with menu cone.
- There will be significant differences in selection time and error rate when participants select with and without continuous visual Feedback.
- There will be significant differences in selection time and error rate when participants select in different directions.

4.2.2.4 Participants

We recruited 15 unpaid participants (11 male and 4 female) ranging between 22 and 30 years of age, with a mean age of 25.4 (sd = 2.41). All participants were postgraduate students. They were all right-handed and had some experience of gestural interaction, such as using a Wii remote or Microsoft Kinect for playing games.

4.2.2.5 Procedure

The task in the experiment was static target selection in a rendered 3D space. We mainly followed the experimental settings in [168] for evaluating a selection task in occluded and densely populated 3D environments. Each trial had a start object, a target and 45 distractors. The diameters of the distracters were between 10 cm and 30 cm and the target diameter was 20 cm. The start object was a yellow sphere, the target was a red cube, and the distracters were grey spheres. The start object was always at the same location in each trial, while the target was at a random location. Six of the 20 cm diameter distractors were placed close to the target in three dimensions, forming a cube (Figure 4-3). Their distance to the target was varied across the DensitySpacing conditions. These distractors were carefully positioned to ensure that the target was visible to the user. In the Occluded conditions, we placed another two of the distractors so that each one partly occluded the target and both together made the target fully occluded. All other distractors across all conditions were randomly distributed in the 3D environment.

Participants pointed the cone to the start object to begin each trial. The start object then disappeared. The time from start object selection to target selection was recorded. To give users a clue to the target location, each trial started with a fade period lasting 1 second during which the target appeared first, followed by the other objects, so the user could see the target location even when the target was subsequently occluded. The distractors appeared blue if selected by the cone. In marker cone, the target changed to green if in range of the cone and became yellow if selected by the depth marker. In menu cone, the target became yellow if it was in range of the cone.

The apex of the cone was fixed at the coordinate origin in the 3D space. In both techniques, the X-Y coordinate of the user's hand in real space was used to control the X-Y coordinate of the cone base center in the 3D space. (All our participants were right handed and chose to use that hand.) To avoid the objects being occluded by the cone, we rendered only the cone center. The distance from the apex to the base center in the Z-dimension was 1200 cm. The target and distractors were distributed within a 300 cm cubic space, 300 to 600 cm from the cone apex in the Z-dimension. To compare the two selection techniques, marker cone also selected 8 objects, and the user needed to move the depth marker along the cone center by moving her hand forward and backward along the Z dimension. The mapping of the movement distance from real space to 3D space was 1:10 for both techniques.

With menu cone, when the user performed the pull gesture to confirm selection, the first scene disappeared and the marking menu was displayed in a vertical plane 200 cm from the ray origin. The user then made a directional gesture to select the target from the marking menu. Following pilot studies, the pull gesture was triggered if the user's hand moved towards her torso faster than 0.5 m/s, and the directional gesture was triggered if the hand moved faster than 0.4 m/s. The "Hand Up" gesture for marker cone was triggered when the user's secondary hand moved directly upwards faster than 0.4 m/s.

Users were instructed to select the target as fast as possible while minimizing their errors. Users had to successfully select the target to end each trial. After the test of each technique, participants were required to finish a computer based NASA TLX(Task Load Index) [77] to measure the cognitive load. NASA TLX is a subjective and multidimensional assessment method for measuring perceived workload, which contains 6 sub scales including Mental Demand, Physical Demand, Temporal Demand, Performance, Effort and Frustration. It is widely used in many studies in various research area [76]. They were also asked to provide feedback at the end. The whole experiment, including introduction of experiment and selection techniques, all trials and breaks, questionnaire, and interviews about feedback, lasted about 60 minutes for each participant.

4.2.2.6 Design

A repeated measures within-participants design was used. For each of the 2 Techniques, 4 blocks of trials were run, one practice block and three test blocks. Within each block, the 12 combinations of TargetDistance, DensitySpacing and Visibility were repeated 3 times, giving 36 trials. The same random order of the 36 combinations of TargetDistance, DensitySpacing and Visibility was used in each trial. The order of presentation of the two techniques (marker cone and menu cone) was counterbalanced; for each pair of consecutive participants, we randomized the order of presentation for the first user

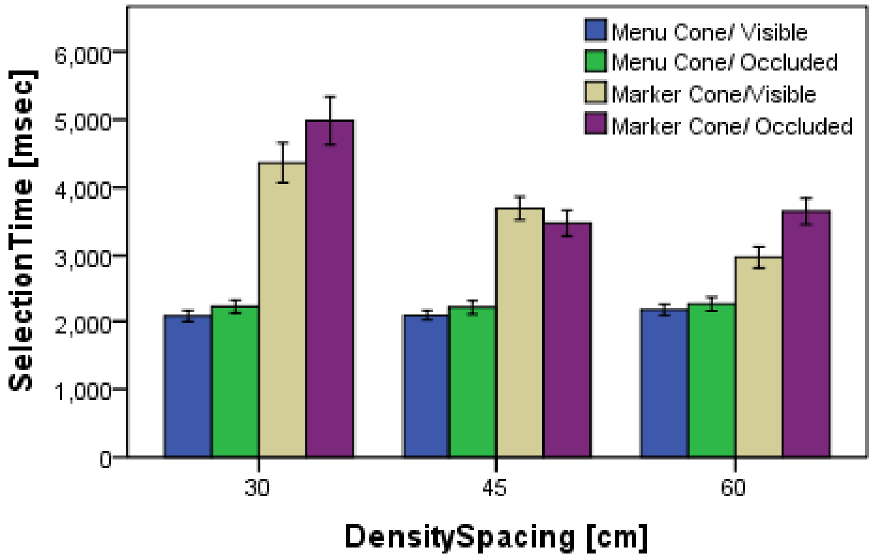


Figure 4-4: *Mean selection time (In this and all later charts, error bars represent 95% confidence intervals)*

and used the other order for the second user. Thus for each participant, 288 trials were performed in total.

4.2.3 Results

In the experiment, the selection time, error rate and task load were recorded, and the collected data are analyzed in this subsection.

4.2.3.1 Selection time

A repeated-measures ANOVA for Technique x TargetDistance x DensitySpacing x Visibility was used to analyze the results. Main effects were found for Technique ($F_{1,14} = 141.06, p < .001$), DensitySpacing [1] ($F_{1.39,19.43} = 48.163, p < .001$), and Visibility ($F_{1,14} = 24.78, p < .001$). TargetDistance had no significant effect ($F_{1,14} = 3.78, p = .07$). Interaction effects were found for Technique x DensitySpacing [1] ($F_{1.33,18.65} = 40.348, p < .001$), and Technique x Visibility ($F_{1,14} = 8.98, p < .05$). No significant interaction effect was found for Technique x TargetDistance ($F_{1,14} = .231, p = .64$). Mean selection time for menu cone was 2.17 s (1.45 s in the selection phase and 0.72 s in the disambiguation phase), and 3.85 s for marker cone.

Post hoc Bonferroni pairwise comparisons showed that menu cone was significantly faster than marker cone ($p < .001$), and selection time for a visible target was less than for an occluded target ($p < .001$). Further analyses using one-way repeated-measures ANOVA for DensitySpacing found that DensitySpacing had no significant

[1]The sphericity assumption was not met so the Greenhouse-Geisser correction was applied; the corrected degrees of freedom are shown.

effect on selection time ($F_{2,118} = 2.83, p = .06$) with menu cone, but with marker cone a significant effect was found [1] ($F_{1.72,101.39} = 65.71, p < .001$). Further analyses using two-tailed dependent T-tests found that an occluded target took significantly longer to select with both menu cone ($t_{89} = 4.23, p < .001$) and marker cone ($t_{89} = 4.00, p < .001$).

4.2.3.2 Error rate

Users were required to select the target successfully. Failure to do so at the first attempt was recorded as an error. For marker cone, an error was recorded if the target was not selected by the depth marker when the user confirmed the selection. For menu cone, an error was recorded if the target was not in the initial selection range when the user made the confirmation pull gesture or when the user failed to select the target in the marking menu. A repeated-measures ANOVA for Technique x Distance x DensitySpacing x Visibility showed no significant difference in error rate between marker cone and menu cone ($F_{1,14} = .32, p = .59$). Mean error rate was 8.21% of all trials for menu cone (with 29% happening in the first phase and 71% in the second) and 8.89% of all trials for marker cone.

4.2.3.3 Task Load

NASA TLX [77] assessments were conducted for the different selection techniques. Average weighted workload score was 49.27 for marker cone and 38.18 for menu cone. Within-subjects analysis of variance showed a main effect of selection technique ($F_{1,14} = 8.46, p < .05$). Bonferroni post-hoc pairwise comparison showed the task load of marker cone was significantly higher than that for menu cone (Figure 4-5).

A one-way ANOVA across the selection techniques found a significant effect on physical demand ($F_{1,14} = 4.70, p < .05$), and effort ($F_{1,14} = 10.70, p < .01$). No significant effect was found on mental demand, temporal demand, performance and frustration. Bonferroni post-hoc pairwise comparisons showed physical demand with marker cone was significantly higher than with menu cone ($p < .05$), and effort with marker cone was significantly higher than with menu cone ($p < .01$).

4.2.4 Discussions

In this subsection we discuss the effects of the selection techniques and 3D environment factors based on the experimental results.

4.2.4.1 Selection Technique

The results show that menu cone outperformed marker cone for freehand gestural selection. On average, selection using menu cone took only 2.17 s, which was only

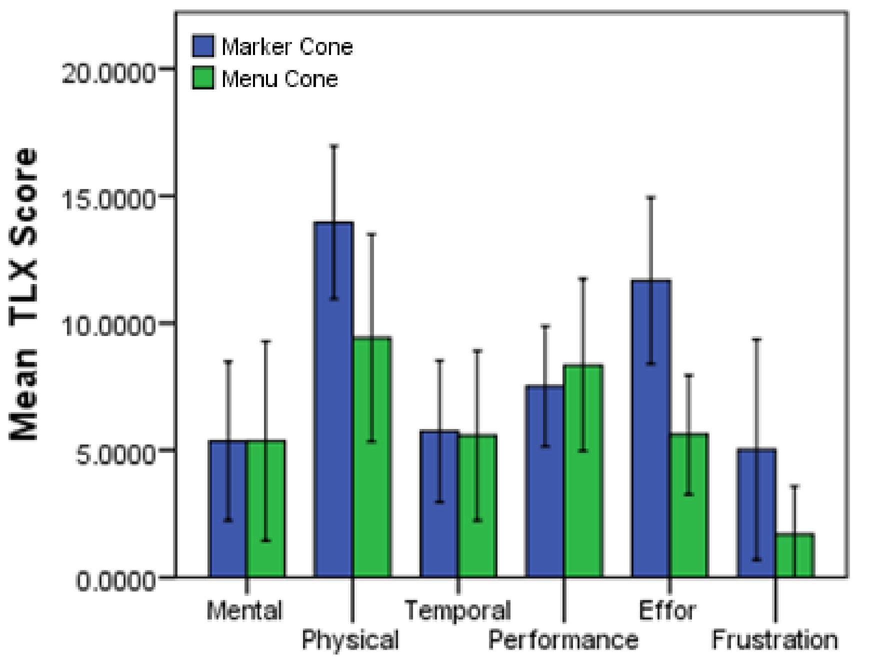


Figure 4-5: *Mean NASA TLX Scores*

56.25% of the time taken with marker cone (3.85 s), with no significant difference in error rates. In Figure 4-4, we can see that menu cone took less time in all density and visibility conditions. The performance difference was also reflected in the post-test questionnaire. All participants said they preferred menu cone. They said it was not easy to control the marker to locate the target with marker cone. Menu cone, on the other hand, was perceived to be more comfortable and quicker.

The task load of marker cone is significantly higher than that of menu cone, especially in physical demand and effort. Users commented that it was tiring to hold their arm in the air to adjust the depth maker to the target. Although menu cone introduced an independent second selection phase, the cognitive demand did not increase, probably because both selection phases were simple and efficient, and were linked together by two connected gestures.

Both techniques use hand movement in the vertical X-Y plane to control cone direction. The difference is in the disambiguation: with marker cone, the user controls the depth marker to disambiguate by moving her hand in the Z dimension; with menu cone, the user uses a pull gesture to confirm selection and a directional gesture to disambiguate in the marking menu. Although marker cone has features of techniques that have proved successful with hand held devices (i.e. depth ray and 3D bubble cursor) [69, 168], users found that it requires precise control so it is not easy to select the target with freehand gestures. Menu cone, on the other hand, requires less precision thanks to the directional gestural marking menu disambiguation, so it is easier to perform and requires less physical effort. Menu cone is also quick because the separate actions of cone selection, pull confirmation and marking menu directional selection are smoothly connected.

Overall, menu cone selection is easier and quicker than marker cone for freehand gestural selection. This suggests that selection techniques with a lower accuracy requirement can be more appropriate for freehand selection tracked by a relatively low resolution remote camera, but need careful design to use smoothly connected gestures.

Our results do not align with previous research [69], which found a marker disambiguation technique (depth ray) to be faster than a marking menu disambiguation technique (flower ray). Such difference may caused by the different gestural tracking methods: the convenience of pressing buttons to trigger a menu or confirm a selection, as used in the other studies with hand held devices, is not available in freehand gestural interaction. Thus we suggest that while the designers of freehand gestural interaction may leverage the designs of other gestural interaction techniques, they should consider carefully whether these designs can be directly carried over to freehand interaction.

4.2.4.2 Environmental Effects

4.2.4.2.1 Density A significant interaction effect on selection time was found for Technique x DensitySpacing. Figure 4-4 illustrates that selection time was stable across different DensitySpacing levels with menu cone, while varying more with marker cone. This may be explained because with menu cone the user only needs to make sure the target is included in the cone selection range, without caring about nearby distractors. However, with marker cone, the user has to move the depth marker closer to the target when the DensitySpacing decreases and the cube selection region becomes smaller, thereby requiring more accuracy and time.

4.2.4.2.2 Visibility Both techniques were affected by target visibility. For marker cone, the whole process is performed in the 3D space so the selection takes longer when the target is occluded and not easy to see. For menu cone, on the other hand, although the disambiguation phase of menu cone is in 2D and not affected by occlusion, the user still needs to select the occluded target in 3D space in phase 1, which contributed more (67%) than the disambiguation phase (33%) to the total selection time, so the total selection time with menu cone was still adversely affected by occlusion.

4.2.4.2.3 Target Distance We found no significant effect for TargetDistance or Technique x TargetDistance. We thought that marker cone would be more affected by TargetDistance because the user's hand would travel a longer distance to select a distant target. However, we observed that when participants selected with marker cone, not all participants started with their hand near and moved it to far. Rather, some extended their arm at the beginning of the trial and moved their hand from far to near to select. These participants selected the distant targets faster than the near targets, so overall there was no significant effect of target distance on selection time.

Although menu cone was faster than marker cone in this study, and there was no significant difference in error rate, there is still room to improve the accuracy of menu cone. Furthermore, when participants performed the directional gestures to select in the menu, they produced different selection times and errors when selecting items in different directions. For example, selection of the item to the right was faster and more accurate than up. Participants also commented that they felt more comfortable gesturing in some directions such as right and down. Some participants suggested that we should give visual feedback of the user's hand position in the disambiguation phase, claiming that the absence of hand position feedback made them lose confidence when making directional gestures. All of these findings led us to explore variations on the disambiguation phase of menu cone in the following studies.

4.2.4.2.4 Comparison with Previous Studies The marker cone is essentially a freehand version of the depth ray technique intended for use with hand-held devices. The results are very similar to those of previous investigations of depth ray selection in dense and occluded 3D environments [168], which suggests that in general similar 3D gestural selection techniques with freehand and hand-held devices bear similar effects from density and target visibility.

Although marker cone is similarly affected as depth ray by the density and visibility of objects in the 3D environment, marker cone is slower and more error-prone than depth ray controlled by a hand held device. This is mainly because the freehand tracking input from a single inexpensive camera currently is not so accurate and because of the absence with freehand interaction of a physical button with which to make inputs (such as confirm) easily. This performance difference is sufficient to make an interaction design that is successful for 3D tracked hand-held devices considerably less successful in freehand 3D interaction. Again, we note that freehand 3D gestural interaction is not the same as 3D gestural interaction using hand-held devices, and designers must beware of assuming that interaction techniques can simply be transferred between them.

4.3 Freehand Text Entry with Virtual Keyboard

The increasing use of interactive TV in the living room or large displays in public space brings novel opportunities and requirements for rich and engaging interactive experiences. There are many different ways to interact with such displays. The most common input method is using a remote or hand-held device, however, such method normally offers only a limited set of buttons and does not lend itself to offering richer means of interaction. Other input methods used for computers or mobile devices can also be used, such as keyboard, mouse, and touch-sensitive displays. However, these input devices are usually installed close to or even contiguous with the screen so they are not suitable for typical use scenarios with an interactive TV where the user is often at a distance from the screen. Mobile devices such as phones or tablets can also be used to interact with remote displays, but configuration is often needed to connect the personal devices so this may not be convenient in some scenarios.

Gestural input is increasingly popular, using hands-on input devices (e.g. Wii Remote) or freehand motion tracking by a camera (e.g. Microsoft Kinect). As gestural input moves beyond home gaming settings, freehand gestural interaction, which has no need for hands-on input devices and so enables easier and more convenient "walk up and use" [13], is likely to become more important in interactions with TV in everyday settings.

The handheld remote control is by far the main input device for TVs, and many text input methods designed for TV remotes have been proposed and investigated [39, 160, 85]. Geleijnse et al. [64] also compared the physical Qwerty keyboard and remote control for text entry and suggested that the Qwerty keyboard is better.

Currently, however, it is still difficult to perform some common tasks such as text entry with freehand gestural interaction. For example, when a person is trying to search for a program or a video clip on an interactive TV, her text entry task may be challenging due to several factors including, for example, the relatively low resolution of many remote gesture sensors and the distance to the TV screen.

Although research has been conducted on text input with various input devices and techniques, most techniques use handheld input devices and so cannot be used directly in freehand interaction. Therefore, we are motivated to investigate freehand gestural text input methods. Here, we report findings from the design and evaluation of some candidate virtual keyboard layouts and input techniques.

4.3.1 Design of Freehand Text Entry

4.3.1.1 Design Consideration

Our aim is to implement text entry methods that facilitate "walk-up-and-use" - or in the case of interactive TV, more likely "sit-down-and-use" - interaction experiences [13], while retaining the simplicity and directness of freehand interaction. Most previous gestural text entry methods use handheld devices or fiducial markers for tracking motion, which can offer accurate tracking of hand, wrist and fingers. Freehand motion tracking enabled by an inexpensive remote camera, on the other hand, can track hand motion robustly but not the small motions of wrists or fingers, especially when users are at a distance from the display/sensor. Besides the lack of fine movement tracking, there are also some other challenges for freehand text input, such as noisy motion tracking, no physical button or surface to click or touch, and no physical support or tactile feedback for the hand.

Text entry methods based on freeform alphabetic character recognition have been investigated in considerable previous work (e.g. [126, 183, 100]). With such methods, however, users need to learn and remember a set of gestures. Such learning demands may not be suitable for scenarios with interactive TV where quick and easy interaction is important. Text entry methods based on word level prediction, such as shorthand writing [195, 101], Swype [2] or text input based on speech recognition [83] are also possible for freehand text entry. But with interactive TV applications, the requirements of non-dictionary word entry could be high (e.g. entering user name, password, email address, or url), thus character based text input may be better suited to interactive TV.

A virtual keyboard can provide easy recognition and learning [92, 155] and, therefore, may be more suitable for interactive TV text entry. Although character arrangement on a virtual keyboard can be optimized according to the context of use and alternative arrangements may improve the performance of expert users [92, 72, 88, 18, 143], the Qwerty layout still has some benefits [196, 155], is the basis of many improved text entry methods [196, 114, 52] and has the advantage of familiarity to many users. For "walk up and use" and entertainment scenarios with interactive TV, the familiarity of the Qwerty layout is very important to users, with less demand for extra learning and less visual scan time. Thus we designed our gestural text entry based mainly on the Qwerty keyboard layout and a character based text entry method.

One benefit of freehand gestural interaction is that the hand can move in 3D space, which means that the virtual keyboard can be in 3D. However, results from previous research [155] indicate that 3D layout text entry has low performance. And previous work on 2D and 3D option selection with freehand gesture [142] also suggests that freehand selection with a 3D layout is less accurate than with a 2D layout. Thus, we designed and evaluated a 2D keyboard layout in this study.

4.3.1.2 Layout

As noted above, Qwerty is a very familiar keyboard layout and has been shown to perform well as a virtual keyboard with a mid-air handheld device [155]. It is therefore a reasonable candidate for freehand text entry and we designed and implemented two virtual keyboard based on Qwerty layout.

4.3.1.2.1 Qwerty The first layout is the standard Qwerty layout. For our prototype design and evaluation, we used a Qwerty layout of 28 characters (26 English

[2]http://www.swype.com/

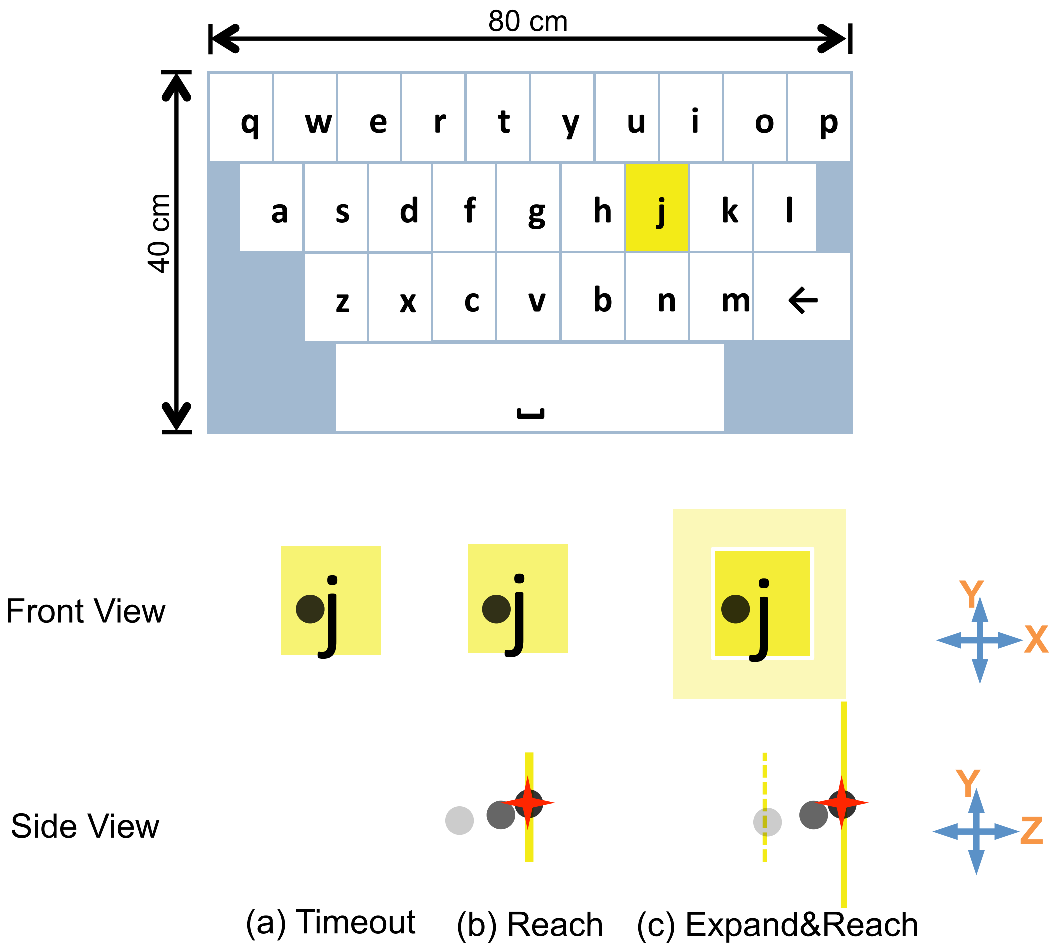


Figure 4-6: *Up: Qwerty layout. Down: Selection techniques; the spherical grey cursor is controlled by the user's hand position.*

letters, space and backspace), similar to the keyboard used for touch-screens such as Windows Phone (Figure 4-6). More keys could also be added in different applications.

4.3.1.2.2 Dual-circle Besides the Qwerty layout, another virtual keyboard layout that has been investigated is circle. Although the circle layout is not as effective as Qwerty with a mid-air handheld device [155], it may bring benefits for freehand interaction. For example, rather than being tiled, characters are arranged to offer easy access to each character from the center of the circle. With accurate handheld devices such as in [155], all characters could be distributed in a single circle. However, for freehand motion tracked by low cost camera, the character size could be too small for reliable use with noisy tracking input in an interactive TV scenario.

To address this issue and to leverage the two-handed operation that freehand gestural interaction allows, we proposed a Dual-circle layout for text entry (Figures 4-7 and 4-8). The characters are evenly distributed in 2 circles next to each other. Thus

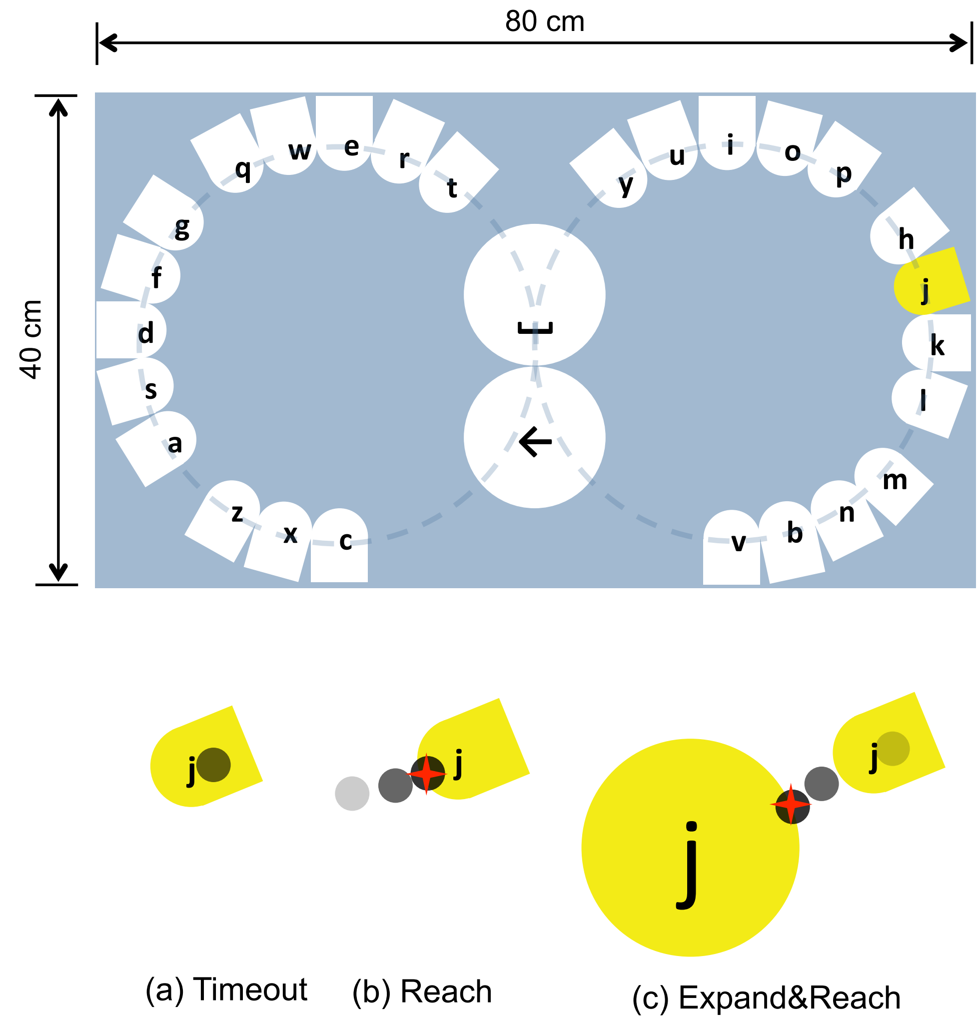


Figure 4-7: *Up: Dual-circle layout. Down: Selection techniques; spherical grey cursor controlled by the user's hand position.*

each character can be bigger than if they were distributed in one similarly sized circle. To leverage users' familiarity with the Qwerty layout, we based the character distribution on Qwerty: the top and bottom of each circle is used for characters located in the top row and bottom row in Qwerty, and the middle row of the Qwerty layout is turned vertically and put sideways in each circle based on the corresponding hand and fingers when using a Qwerty keyboard. Gaps are used to separate the different character groups for a clearer mapping to the familiar Qwerty layout.

Space and backspace are represented by circles located in the middle of the keyboard for easy access with both hands. As users dislike selecting in the left-down direction with the right hand [142], the left-down portion of the right circle and the mirrored portion of the left circle are left blank.

4.3.1.3 Character Selection

As noted above, without a physical device in the hand and buttons to click, freehand gestural selection can be challenging. And although finger and wrist movement are used in some previous work, they are less suited to freehand interaction tracked by a remote inexpensive camera, so other techniques are required.

4.3.1.3.1 Timeout One common method for freehand selection is pointing to the target and waiting for a timeout threshold. It is easy for novices to understand and perform and is accurate for large targets. The primary disadvantage is that the dwell time can slow overall selection time. For both our layouts, the timeout selection method can be used: the user points to a character by the X-Y position of her hand and waits for the timeout threshold, e.g. 1.2 s, to select the character (Figure 4-6.a, Figure 4-7.a).

4.3.1.3.2 Reach As an alternative to timeout, hand motion can also be used for selection confirmation. The Reach technique has been used for target selection with freehand gestures [141], in which users select by moving their hands to reach into the target in 3D. For a virtual Qwerty keyboard placed vertically in front of the user's body position (in this case at 40 cm), the user can point to the desired character by X-Y movement of the hand and then reach forward in the Z-dimension to select the character (Figure 4-6.b). For the Dual-circle layout, the character can be selected by moving the hand's X-Y position to reach across the border of the desired character tab (Figure 4-7.b).

With the Reach technique, although 3D hand position is required in the Qwerty layout while only X-Y position is required in the Dual-circle layout, with both layouts the user's hands move freely in 3D space to reach the characters. In practice, with both Qwerty and Dual-circle layouts, the user tends to move the selecting hand forward and towards the target character simultaneously in one fluid movement.

4.3.1.3.3 Expand&Reach In both layouts, the character size is relatively small so could be difficult and error-prone for freehand selection. It is also difficult to expand the target in motor space for tiled targets in 2D interfaces [122]. However, since the hand can move in 3D space, the combination of the extra dimension and the Reach selection technique brings new interaction opportunities. We designed additional Expand&Reach techniques for both keyboard layouts. With the Qwerty layout, when the user points to a character, an expanded target appears along the Z-dimension (e.g. 5 cm further away than the current hand position and 2 times bigger than original size). The user moves her hand forward to reach the expanded target in order to select (Figure 4-6.c). With the Dual-circle layout, when the user points to a character, an expanded character tab containing the target character appears in the center of the

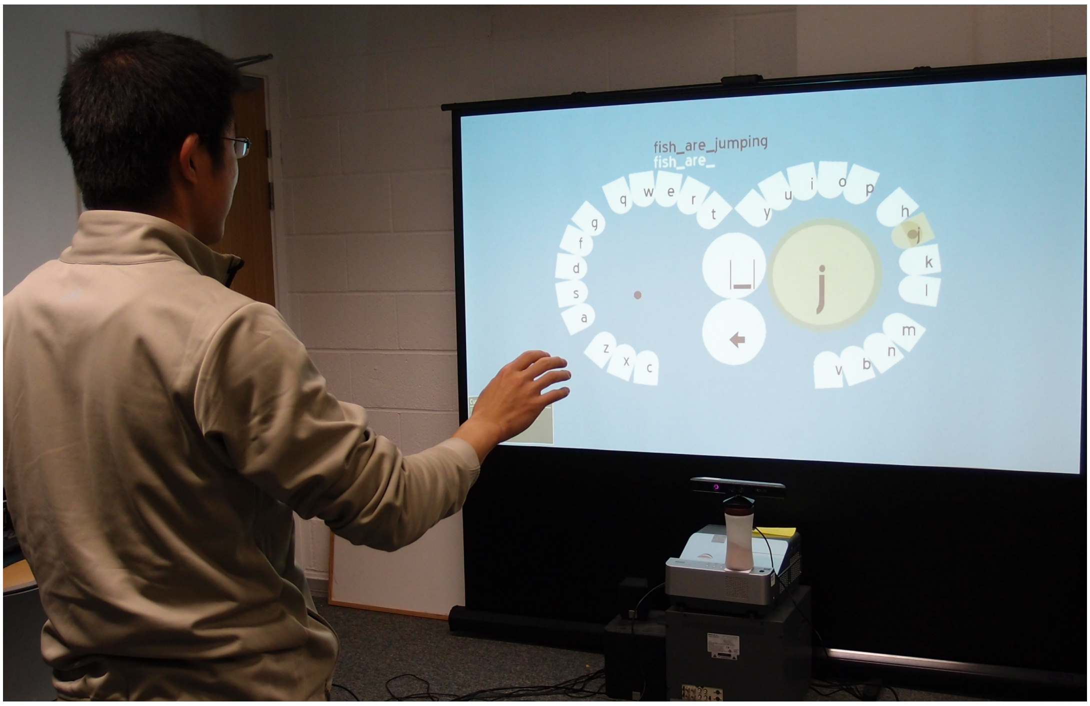


Figure 4-8: *Experimental setting.*

corresponding circle (Figure 4-8). The user moves her hand to reach the expanded target to select it (Figure 4-7.c).

There are several potential advantages of the Expand&Reach technique:

(i) easier selection - the target expands in both visual and motor space;

(ii) error tolerance - users need to move their hand to reach the expanded target to confirm selection, so if they notice a selection error before reaching the expanded target, they have a chance to (re)select the right character;

(iii) requires only hand position tracking - no fine finger movement or posture tracking is needed.

4.3.2 Experimental Evaluation

A controlled experimental evaluation was conducted to investigate the effects of the different keyboard layouts and selection techniques.

4.3.2.1 Equipment and Setting

A Sanyo PDG-DWL2500 3D projector was used at 1280 x 720 resolution with a 203 x 115 cm screen centered at 130 cm height to simulate a large interactive TV display. A Microsoft Kinect camera was used with a refresh rate of 30 fps and the Kinect for Windows SDK V1.5 on Windows 7. The Kinect camera was placed 50 cm in front of the screen at a height of 70 cm. The user stood 250 cm from the screen (Figure 4-8).

4.3.2.2 Independent Variables

The independent variables were Layout (Qwerty, Dual-circle), Selection Techniques (Timeout, Reach, Expand&Reach), and Day (1 to 5).

4.3.2.3 Hypotheses

- There will be significant differences in typing speed and error rate when participants input with Qwerty vs with Dual-circle layout.
- There will be significant differences in typing speed and error rate when participants input with Timeout, Reach, and Expand&Reach.
- There will be significant differences in typing speed and error rate when participants input on different days.

4.3.2.4 Participants

6 participants (4 males, 2 females) were recruited from the local campus, mean age 26 (sd = 1.7), all right handed and with some experience of gestural interaction for gaming.

4.3.2.5 Procedure

Each character is white and highlights yellow when pointed at. With Timeout selection, the colour gradually changes from yellow to green until timeout. Using Reach and Expand&Reach with Qwerty, the character gradually changes from yellow to green as the hand moves forward to reach it. The selected character appears immediately below the target sentence. A “typing” sound is played when a character is entered correctly. If the entry is incorrect, an error sound is played instead and the input is shown in red. All mistakes must be corrected for each sentence.

Both keyboards were 80 cm x 40 cm, with the top edge at the same height as the user’s shoulders in motor space. Two spheres sized 2 cm were controlled with the hands. With the Qwerty layout, the hand in front of the other is enabled and rendered in black, the other is rendered grey and disabled to avoid accidental selection. With the Dual-circle layout, the blank center area is large enough to accommodate an idle hand without accidental selection, so no disable mechanism was used. For timeout selection technique with both layouts, the dwell time was 1.2 s. The 1.2 s dwell time was based on a pilot study which showed that less than 1.2 s produced more errors, and the observation that almost all commercial Kinect interfaces with timeout selection use more than 1.5 s.

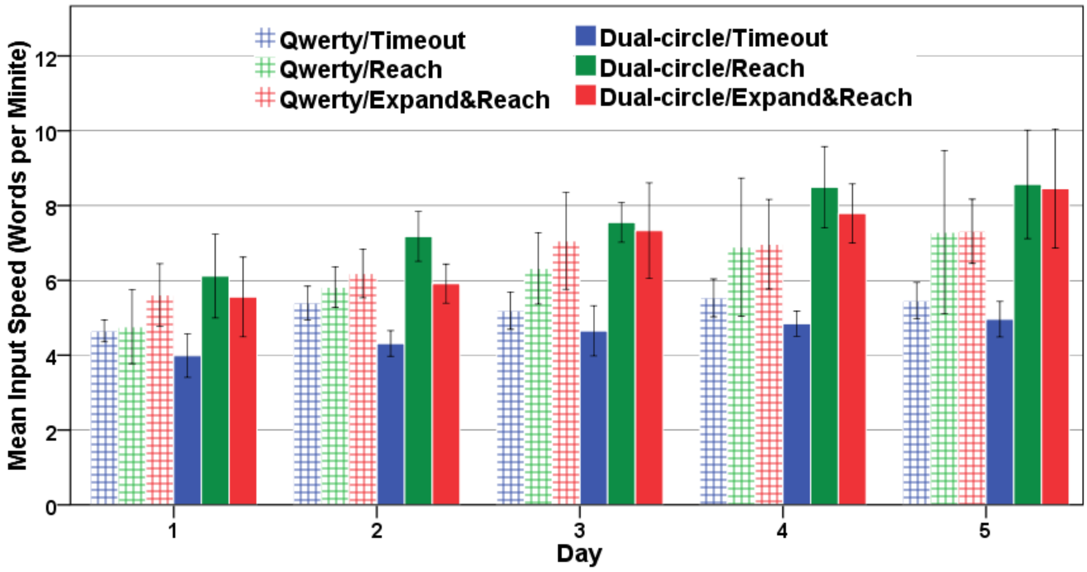


Figure 4-9: *Mean text input speed.*

4.3.2.6 Design

A repeated measures within-participants design was used. The evaluation lasted for 5 days with 6 sessions every day. Each session tested a combination of Layouts and Selection techniques. In each session, 4 sentences were presented for the participant to reproduce, the first as practice followed by 3 test sentences. Six sets of sentences were randomly selected from MacKenzie and Soukoreff's phrase sets [116] and were assigned randomly to different sessions. User preferences and NASA task load index (TLX) data were collected on the first and last days.

4.3.3 Results

4.3.3.1 Typing Speed

A repeated-measures ANOVA for Layout x Selection Technique x Day was used to analyse the text input speed. Main effects were found for Selection Technique ($F_{2,10} = 144.27, p < .001$) and Day ($F_{4,20} = 21.49, p < .001$). Layout had no significant effect ($F_{1,5} = 1.38, p = .29$). Interaction effects were found for Layout x Selection Technique ($F2, 10 = 11.69, p < .01$) and Day x Selection Technique ($F_{8,40} = 6.05, p < .001$).

Post hoc Bonferroni pairwise comparisons showed that Reach and Expand&Reach were both significantly faster than Timeout ($p < .001$), with no significant difference between them. Text input speed in the last 3 days was significantly faster than on the first day ($p < .05$), with no significant difference between the last 3 days. Mean text input speeds across all conditions are shown in Figure 4-9 and Table 4.1.

	Qwery			Dual-circle		
	Time-out	Reach	Expand&Reach	Time-out	Reach	Expand&Reach
Day 1	4.65	4.76	5.61	3.99	6.11	5.56
Day 2	5.40	5.82	6.19	4.31	7.17	5.91
Day 3	5.19	6.32	7.06	4.65	7.55	7.33
Day 4	5.53	6.89	6.97	4.84	8.49	7.79
Day 5	5.46	7.29	7.31	4.96	8.57	8.46
5 days overall	5.25	6.22	6.63	4.55	7.58	7.01

Table 4.1: *Mean speed (wpm) over 5 days.*

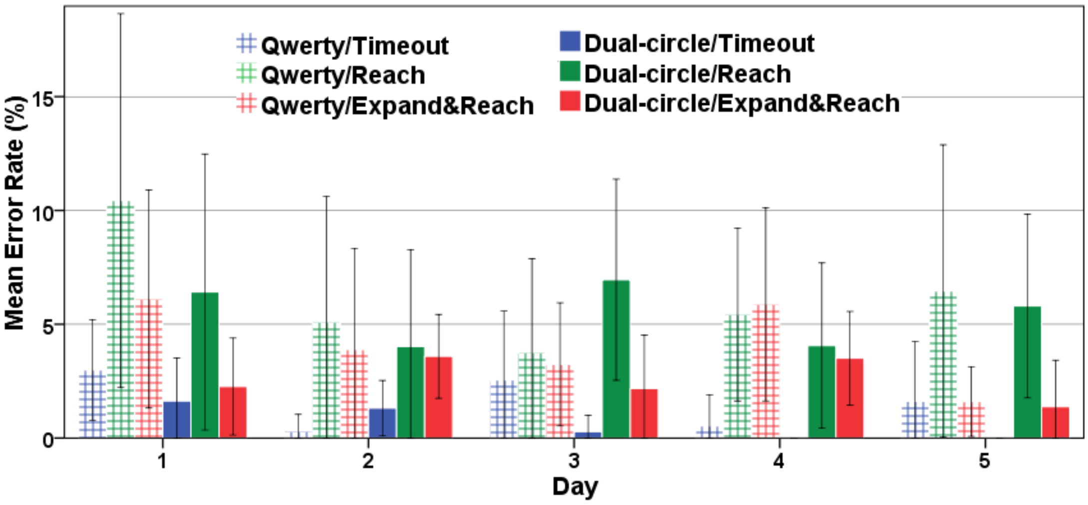


Figure 4-10: *Mean error rate.*

4.3.3.2 Error rate

A repeated-measures ANOVA for Layout x Selection Technique x Day was used to analyze error rate. Main effects were found for Layout ($F_{1,5} = 10.86, p < .05$) and Selection Technique ($F_{2,10} = 11.09, p < .01$). Day had no significant effect ($F_{4,20} = 1.94, p = .14$). No interaction effect was found. Mean error rates are shown in Figure 4-10 and Table 4.2.

Post hoc Bonferroni pairwise comparisons showed Qwerty to be significantly more error-prone than Dual-Circle ($p < .05$). Timeout had significantly fewer errors than Reach ($p < .05$). There were no other significant differences between selection techniques.

4.3.3.3 Hand Movement in 3D Space

We also recorded the hand movement distance per character in day 5. A repeated-measures ANOVA for Layout x Selection Technique was used to analyze hand movement distance per character. Main effects were found for Layout ($F_{1,5} = 39.42, p < .01$) and Selection Technique ($F_{2,10} = 10.21, p < .01$). No interaction effect was found.

	Qwery			Dual-circle		
	Time-out	Reach	Expand&Reach	Time-out	Reach	Expand&Reach
Day 1	3.00	10.44	6.11	1.63	6.42	2.27
Day 2	.29	5.10	3.89	1.32	4.02	3.59
Day 3	2.55	3.75	3.24	.28	6.98	2.18
Day 4	.53	5.43	5.88	.00	4.06	3.51
Day 5	1.61	6.46	1.60	.00	5.81	1.38
5 days overall	1.60	6.24	4.14	0.64	5.45	2.58

Table 4.2: *Mean error rate (%) over 5 days.*

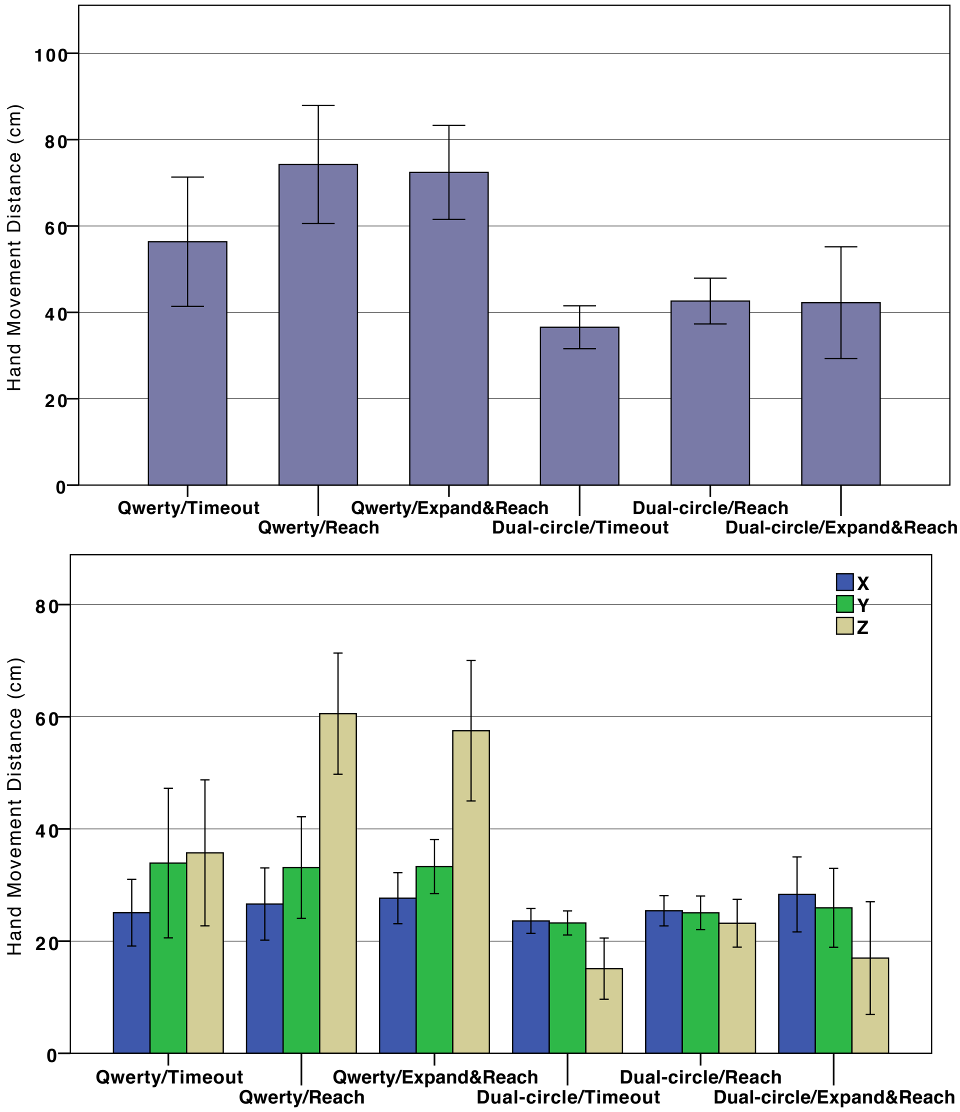
Post hoc Bonferroni pairwise comparisons showed that Qwerty layout required significantly more hand movement distance than Dual-Circle ($p < .01$). The Timeout selection technique had significantly less movement distance than Reach ($p < .05$). There were no other significant differences between selection techniques. Mean hand movement distance per character is shown in Figure 4-11(a).

We analysed the hand movement distance in different axes (i.e. X, Y, Z) using one-way ANOVA for six text entry methods. We found that with Qwerty/Timeout and Dual-circle/Reach there was no significant effect of axis ($p < .05$). With Qwerty/Reach, Qwerty/Expand&Reach and Dual-circle/Timeout, main effects were found for Axis ($p < .001$). Post hoc Bonferroni pairwise comparisons showed that with Qwerty/Reach and Qwerty/Expand&Reach methods, hand movement in the Z axis was significantly more than in the X and Y axes ($p < .05$), while with Dual-circle/Timeout and Dual-circle/Expand&Reach methods, hand movement in the Z axis was significantly less than in the X and Y axes ($p < .01$). Mean hand movement distance per character in the X, Y and Z axes is shown in Figure 4-11(b).

4.3.3.4 Task Load

A repeated-measures ANOVA for Layout x Selection Technique x Day was used to analyse the NASA task load index (TLX). Main effects were found for Layout ($F_{1,5=9.21}, p < .05$), Selection Technique ($F_{2,10=}5.75, p < .05$) and Day ($F_{1,5} = 7.61, p < .05$). No interaction effects were found.

Post hoc Bonferroni pairwise comparisons showed that the Qwerty layout had significantly higher task load than the Dual-circle layout ($p < .05$), the Reach selection technique had significantly higher task load than Expand&Reach ($p < .05$), and the task load on day 1 was significantly higher than on day 5 ($p < .05$). Figure 4-12(a) shows the overall workload. Figures 4-12(b) to (g) show users' mental demand, physical demand, temporal demand, performance, effort and frustration.



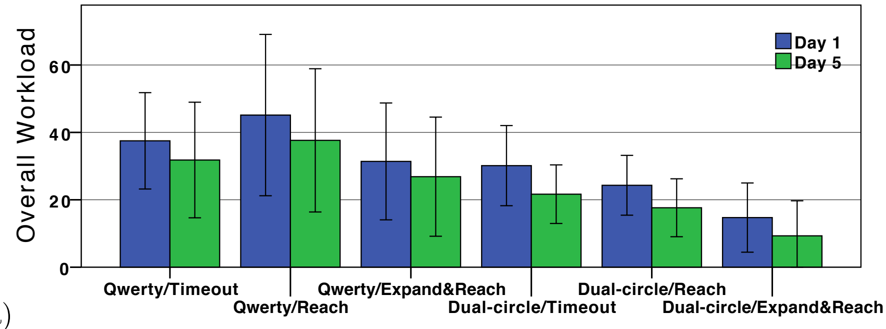
(a)

(b)

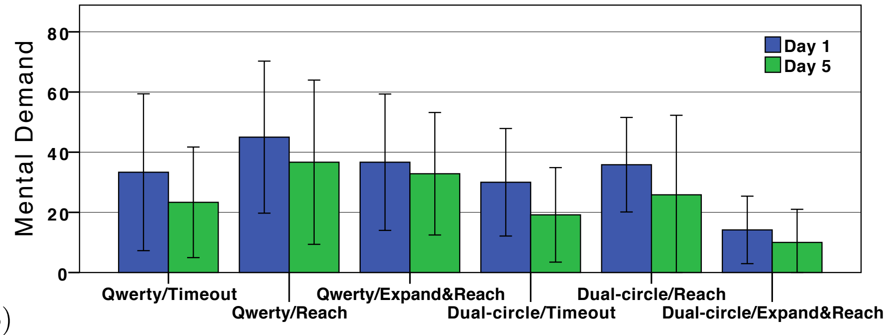
Figure 4-11: *Mean hands movement distance per character. (a) Hand movement distance in 3D space (b) Hand movement distance in X, Y and Z axis.*

4.3.3.5 User Preference

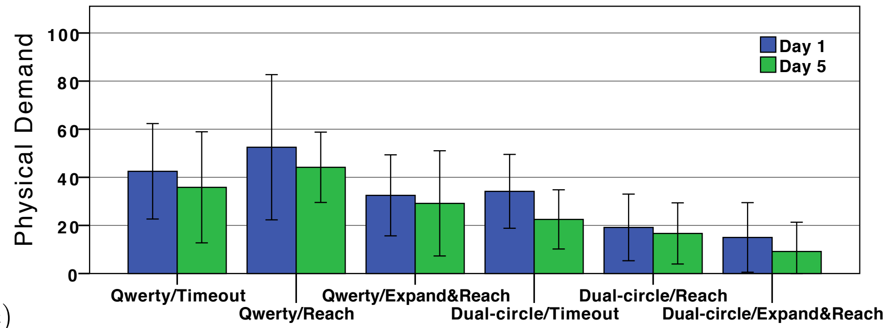
User preference data for each input method was collected on the first day and the last days. Users gave their preference (from 1 for strongly dislike to 10 for strongly like) after each text entry method, as shown in Figure 4-13. Overall, all participants preferred Dual-circle/Expand&Reach on both the first and last days of the study. The Dual-circle/Expand&Reach technique could enable people to input text comfortably in both "walk up and use" and the slightly longer term use (5 days) of our study. Its error tolerance also allows more casual hand movements without high concentration and physical effort.


Overall Workload
0
20
40
60
Day 1
Day 5
Qwerty/Timeout
Qwerty/Reach
Qwerty/Expand&Reach
Dual-circle/Timeout
Dual-circle/Reach
Dual-circle/Expand&Reach

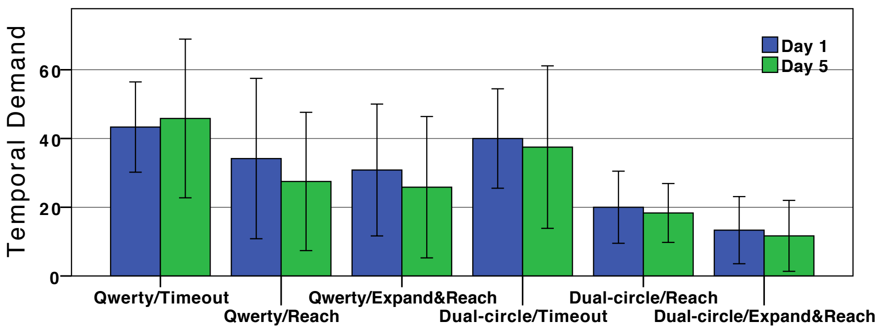
(a)


Mental Demand
0
20
40
60
80
Day 1
Day 5
Qwerty/Timeout
Qwerty/Reach
Qwerty/Expand&Reach
Dual-circle/Timeout
Dual-circle/Reach
Dual-circle/Expand&Reach

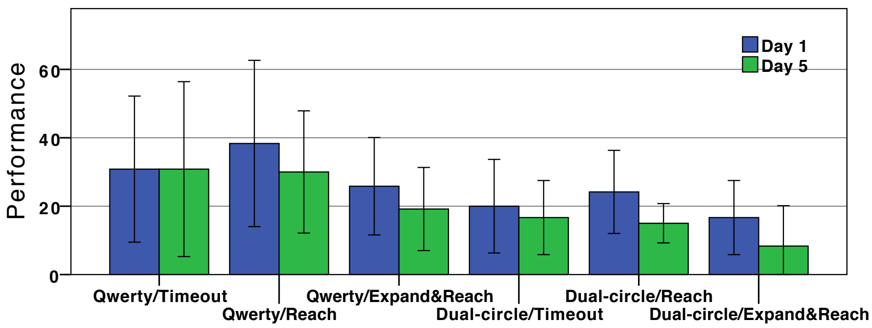
(b)


Physical Demand
0
20
40
60
80
100
Day 1
Day 5
Qwerty/Timeout
Qwerty/Reach
Qwerty/Expand&Reach
Dual-circle/Timeout
Dual-circle/Reach
Dual-circle/Expand&Reach

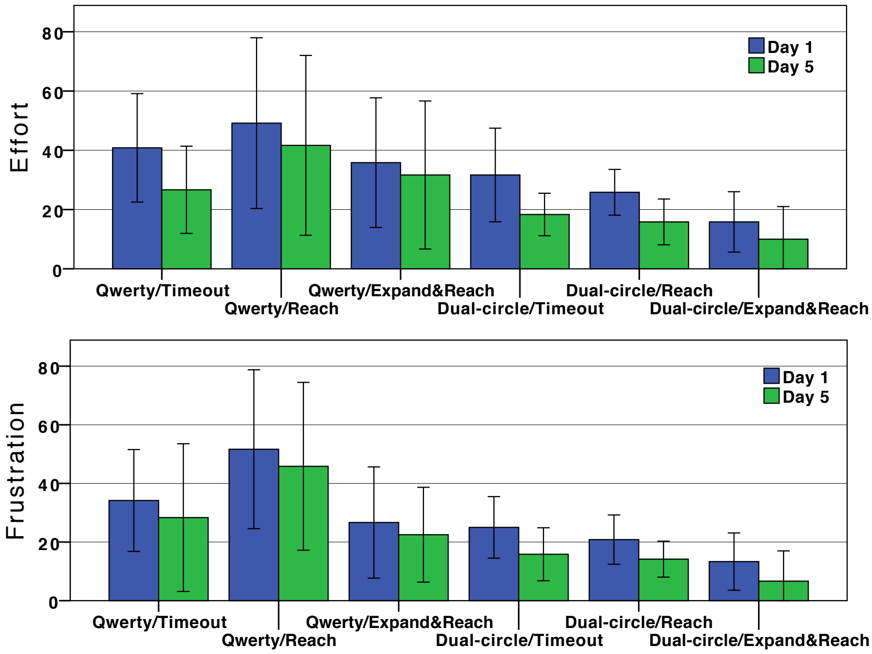
(c)


Temporal Demand
0
20
40
60
Day 1
Day 5
Qwerty/Timeout
Qwerty/Reach
Qwerty/Expand&Reach
Dual-circle/Timeout
Dual-circle/Reach
Dual-circle/Expand&Reach

(d)


Performance
0
20
40
60
Day 1
Day 5
Qwerty/Timeout
Qwerty/Reach
Qwerty/Expand&Reach
Dual-circle/Timeout
Dual-circle/Reach
Dual-circle/Expand&Reach

(e)



(f)

(g)

Figure 4-12: *NASA task load index (TLX) results. (a) Overall workload (b) Mental demand (c) Physical demand (d) Temporal Demand (e) Performance (f) Effort (g) Frustration.*

4.3.4 Discussions

4.3.4.1 Text Entry Method

The dual-circle layout had better performance and lower task load than the Qwerty layout. Although users are more familiar with the Qwerty keyboard, its characters are packed in 2D, so character selection is more difficult than with the dual-circle layout. We noticed that when users select a character with one hand, they normally put down the other hand to avoid careless error selection. With the dual-circle layout, they can just relax the hand because there is enough blank space in the center to prevent careless error selection.

When using the Qwerty/Reach text entry method, users felt it was difficult to find the reach point in the beginning due to the absolute character position. High physical demand was also reported for Qwerty/Reach (Figure 4-12c). With Qwerty/Expand&Reach, on the other hand, as relative location was used, it was easier to select. However, both Qwerty/Reach and Qwerty/Expand&Reach required long movement distance when inputting text (Figure 4-11a), which was largely due to the hand movement for character selection along the Z dimension (Figure 4-11b). In contrast, as no hand movement along the Z dimension was required with the dual-circle layout, and users felt more easy and relaxed in their text entry tasks.

Participants also noticed their improvement for the Reach and Expand&Reach techniques with both keyboard layouts. Especially with the Dual-circle layout, typing speed

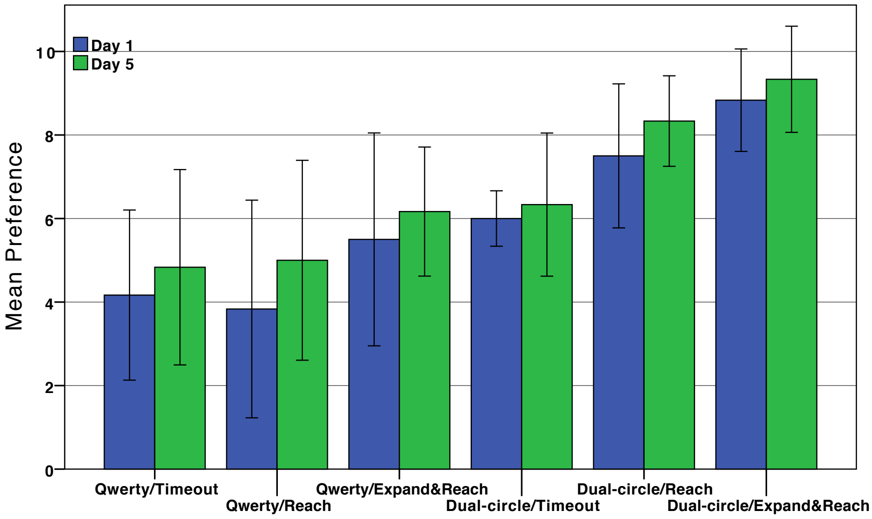


Figure 4-13: *User preference*

increased continuously over 5 days (Figure 4-10). Timeout selection had few errors with both layouts, but the dwell time slowed text entry speed so it was not liked by users. Typing speed with the timeout method, with both layouts, did not improve in the course of the study as other text entry methods did.

Overall, Dual-circle/Expand&Reach was the best text input method. It is error tolerant thanks to the Expand&Reach selection technique so users make fewer careless wrong selections. The dual-circle layout also leaves enough blank space in the center to help people to relax their hands when not inputting text without worrying about triggering a selection when relaxing the hand. Thus Dual-circle/Expand&Reach is a practical text entry method for interactive TV. The Dual-circle/Reach method is more straightforward for selection but was found to be error-prone in this study. It could be a fast entry method given future improvements in tracking resolution. Qwerty/Expand&Reach could be used in scenarios in which familiarity with the standard Qwerty keyboard is desirable. The Qwerty/Reach method, however, may be a less promising method, given its high error rate and the high mental and physical effort noted by the participants.

4.3.4.2 Timeout Dwell Time

Typing speed with the timeout method, with both layouts, did not improve with practice. As noted above, the 1.2 s dwell time was based on a pilot study that showed a dwell time of less than 1.2 s produced more errors for beginners, and the observation that most commercial Kinect timeout based interaction uses more than 1.5 s. Our results suggest that the 1.2 s dwell time was suitable for first time use, however, it could be unnecessarily long for more experienced users, thus limiting performance. We also found that error rates using the timeout selection technique were low. With Dual-circle/Timeout, there were no errors at all in the last two days (Figure 4-10). Users also commented that the dwell time for the Dual-circle/Timeout method could be shorter after some practice.

For the dual-circle layout, it is possible to reduce the dwell time to any value. For example, the dwell time could be reduced even to zero, and in this extreme case, the Dual-circle/Timeout method will be equivalent to the Dual-circle/Reach method, leading to faster typing speed but more errors. With the Qwerty layout, however, the dwell time cannot be reduced so much. This is because users must move their hands over other characters to select the desired character in the Qwerty layout, thus a very small dwell time would trigger the undesired selection easily and lead to a huge increase in errors.

In a previous study using eye gaze for text entry, Majaranta et al. [117] showed that adjustable dwell time could improve text entry performance with a Qwerty layout keyboard, and this could also be the case for freehand text entry. However, in their experiment, participants used a 282 ms dwell time in the last session [117], which may be too short for a Qwerty layout with freehand gestural input. Further investigations of optimal dwell times for freehand interaction could form part of future work.

4.3.4.3 Tracking Sensors and Freehand Gestural Text Input

Our work aims to facilitate freehand gestural interaction tracked by a low-cost remote single camera with no requirement for a user to carry, wear or pick up an input device or fiducial markers. We used Microsoft Kinect in this study as it is a typical currently available low-cost remote gesture tracking sensor. Some more accurate tracking systems, such as Vicon cameras, can offer high accuracy but they require markers on the body and calibration before use, which we explicitly want to avoid. Other sensors, such as the soon to be available LeapMotion sensor, may provide accurate hand tracking without markers, but the tracking range is relatively short and thus may not be suitable for interactive TV.

Even with remote single tracking sensors having higher resolution and become more affordable, there are still some common limitations of freehand interaction that cannot be solved by increasing tracking resolution: e.g.

- having only one tracking angle brings occlusion of finger/wrist motion.
- previous work has shown that when the hands are moving freely in 3D, the user cannot point as accurately as with a 2D surface [165].
- motor control with large muscle groups, e.g. the shoulder, is less accurate than with small muscle groups, e.g. fingers [68, 142]. Even with technical advances, remote camera tracking will share many of the current benefits and limitations

and our main findings are broadly applicable across this class of devices. It is highly unlikely that improvements in tracking resolution would have a detrimental effect on Expand&Reach. Dual-circle/Reach, as suggested above, although error-prone in this study, could be a fast entry method given future improvements in tracking accuracy.

The performance of Dual-circle/Expand&Reach (5.56 wpm on the first day, 8.46 wpm on the last day and a mean of 7.01 wpm over 5 days) also compares well to similar gestural text input without word prediction, such as a mean of 5.4 wpm after 4 days' practice using accelerometer sensors in hand held devices [92], a mean of 6.5 wpm over 2 weeks with a data glove and fiducial trackers [126], and a mean of 5.18 wpm with a text entry method combining speech and gesture [83]. The performance of Dual-circle/ Expand&Reach is also much better than the 1.83 wpm text entry speed reported with the default Xbox gestural text input [83]. On the other hand, the performance of Dual-circle/Expand&Reach achieved in our study still cannot compete with some text entry methods enabled by more accurate tracking devices or input models, such as a Qwerty virtual keyboard and handheld devices (18.9 wpm) [155], or text entry using gaze (nearly 20 wpm) [117].

4.4 Design Suggestions

Interestingly, previous work using hand held devices for text input showed that the Qwerty layout was faster and had fewer errors compared to a circle layout when the users were about 2.5 m away from the display [155]. In our study, on the other hand, the results suggested exactly the opposite. Such differences are largely due to differences in user behavior with different input methods. Moving the bare hand and arm freely in the air is different from holding and moving a mouse on the desktop, tapping the thumb on a mobile phone, or multi-touch movements with the fingers on a touch sensitive surface. For example, in [155], the desired character is selected by pointing and pressing a button using a Wii remote, however, in freehand gestural interaction button pressing is not available and so must be replaced by other techniques, such as Timeout, Reach, and Expand&Reach.

From the findings of our study, some design guidelines emerge for freehand interaction.

- It is not appropriate to require users to perform single actions with high accuracy, or to keep their hands in the air for a long time. This is mainly because the hand is moving without any physical support, so these actions will make the arm fatigue more quickly. One possible solution could be to separate a highly demanding interaction task into several connected low demand tasks. This could not only reduce the physical demands, but also potentially achieve better performance. Furthermore, it is also useful to provide a means to rest the user's arm if the task requires some time to finish.

- Although the hand can move freely in 3D space, and hand motion in the Z dimension could be used as a gesture delimiter or trigger [142], frequent movements in the Z dimension are not recommended for freehand gestural interaction. This is not only because the hand moves more slowly forward and backward [68], and actions in the Z dimension can be error-prone [142], but also because frequent movements in the Z dimension can increase the hand movement distance and corresponding physical demands, as shown in this study.

- The circle layout is useful for interface design with freehand gesture. In the circle layout, all items have one side facing to the center and, therefore, can be reached directly without moving the hands over other objects. And the large blank area in the center can allow users to relax their hands without worrying about triggering undesired actions, thus potentially reducing arm fatigue.

- Combining the expanding target and reach selection techniques is useful for freehand interaction, especially for hand motion tracking by inexpensive sensors with low resolution. When used with the reach technique, the target can expand not only in virtual space but also in motor space. Thus, using the expanding target and reach techniques together addresses the limitation that targets can expand only in virtual space [122], and it is also well suited to freehand interaction techniques due to its error tolerance.

4.5 Design Directions and Examples for Freehand Selection

According to the investigations in this and the previous chapter. We can find out that Reach selection method could be a suitable technique for freehand interaction. However, the noisy input from a single remote camera limits its ability with small targets, and the goal crossing selection technique has also been found to perform better for large and close targets [2]. Thus, if the targets are small, it is difficult to select the target effectively with the freehand Reach technique alone, and so the Reach technique could be improved further to support broader application areas.

4.5.1 Combining Reach, Expanding Target and the Extra Dimension for Freehand 3D Selection

Currently most commercial products (e.g. Xbox Kinect) use large icons and time thresholds for selecting with freehand gestures, requiring the cursor to stay within the target icon for a certain time to select. This method normally demands large icons as users need not only to point to them, but also to stay inside the target for a while. Hence, it is not easy to select small targets with this method. The time threshold could also introduce selection delay and fatigue.

Small target selection with freehand gestures is difficult not only because of the target size, but also the low accuracy of hand tracking. To address these issues, the menu cone technique in the first study here used a method to expand the small and densely concentrated targets to a larger and more structured interface for easier selection. This "expand and select" method is useful for freehand 3D interaction, especially when the targets are small.

However, although expanding targets is a popular topic in 2D interface design[122], the experience with pointing devices or touch surfaces cannot simply be ported directly to freehand interaction because of the lack of button clicks or surface taps. Furthermore, the expanding target selection in 2D interfaces with the point-and-click method has its own limitations. For example, expanding targets when the cursor approaches can magnify only in visual space but not motor space when the targets are closely packed [122]. Although a predictor could be used to increase the motor space before the cursor enters the target area, the benefit is very limited [122]. A 2D fisheye menu is also not as effective as a hierarchical menu, and may increase cognitive load [81].

To design more usable and effective 3D freehand selection, we should try to leverage the benefits of freehand movement (e.g. movement in 3D space), and try to avoid the disadvantages compared to traditional desktop 2D interaction (e.g. low tracking accuracy, no button click). First, as discussed above, the Reach technique is a suitable selection technique for freehand 3D interaction and could be extended. Secondly, although the target expanding selection technique has limited power in a 2D interface with mouse point-and-click, it could be more effective with freehand 3D interaction and the Reach technique. Furthermore, as the hand can be moved and tracked in 3D space, the target expanding and selection are not limited to a 2D plane, but can be extended to 3D space. The integration of the Reach technique, expandable targets and freehand 3D interaction could create powerful new tools for 3D freehand gestural selection.

To combine the Reach technique and expanding target with freehand 3D interaction, the user can move the hand to the target, and the target then expands in a certain direction to create a reachable boundary. The user then confirms the selection by reaching to the expanded boundary to complete the selection. In contrast to selecting with a mouse by pointing-and-clicking inside a 2D target, reaching the expanding boundary of a target can take place outside the original target position in 2D or 3D space.

In the following subsection, we introduce some example interaction designs based on this concept.

4.5.2 Interaction Design Examples

For the selection tasks, the targets could be single isolated objects, or objects densely packed in 1D, 2D or 3D space, as well as in different sizes. Although different types of expanding and distortion in 3D are discussed for information visualization [36], there are few discussions of using the expanding target for selection in 3D. To address different scenarios and applications for freehand selection, we illustrate some examples in Figure 4-14.

4.5.2.1 Basic Selection without Expanding

Figure 4-14.1 shows an example of basic 2D freehand menu control with few items, so no expanding is required in this case. We use the volume control of a TV or multimedia player as an example as it is a frequently used function but not easy for freehand gesture control [43]. Users reach the side border of the display to invoke the menu (a), which is two orange borders. Users then reach the border to select and control the volume (b). If the user reaches across the border and stays, the volume change will be continuous. Users can cross the display border again to hide the menu (c). The same method can be used for similar interactions to support selection with few and large targets, such as TV channel increase/decrease, or picture collection navigation. If more options are required, the menu layout in 2D option selection (Figure 3-1a) could be considered.

Although similar graphic designs are available in some interactive TVs (e.g. Samsung Smart TV) with freehand gestural control, hand poses (e.g. “hand open” to move the cursor and “hand close” to “click”) are typically required to simulate desktop interaction. Considering that the design in Figure 4-14.1 can be performed without any hand pose, and that crossing outperforms pointing for large or proximate targets [2], this design could make the simple selection tasks (e.g. volume and channel control) easier and faster. Given that these are among the most frequently used functions of a TV, this design could make freehand control for interactive TV substantially more usable.

4.5.2.2 Targets Packed in 1D

Figure 4-14.2 shows a example of selection with expanding targets packed in one dimension. The menu items are in a vertical layout, and when the hand moves into one menu item, it expands (the light blue rectangle) and the user can reach for the right

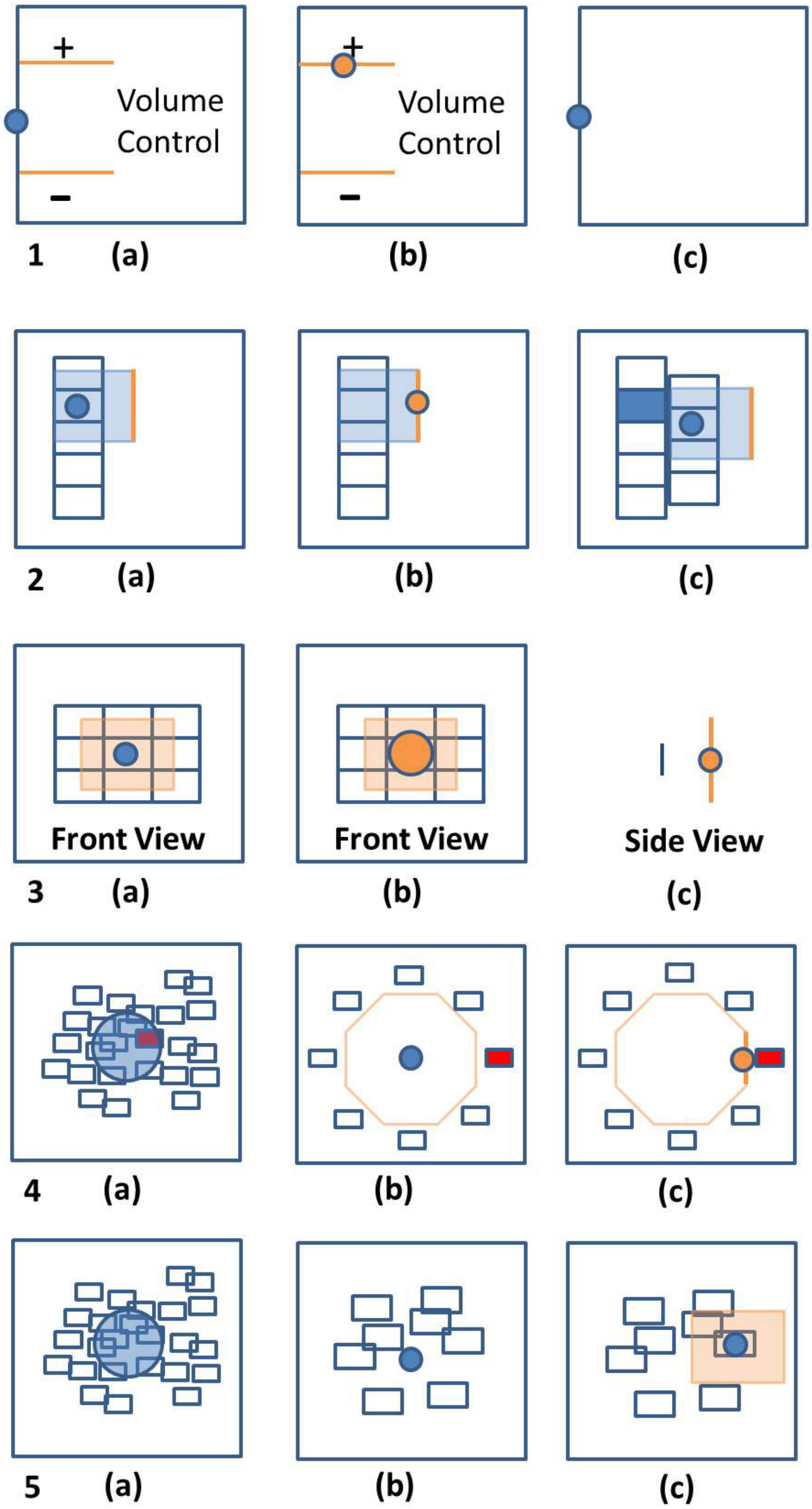


Figure 4-14: *Design examples for freehand selection. The sphere is the cursor following the hand movement. Its color turns from blue to orange indicating its reaching the target boundary. The orange object is the target. (1) Simple selection with few targets (2) Targets packed in 1D (3) Targets packed in 2D (4) Targets in dense and occluded 2D or 3D environment with different appearance (5) Targets in dense and occluded 2D or 3D environment with similar appearance.*

border (the orange line) of the expanded area to select this target (a). When the hand reaches the border, the item is selected (b). If it is a hierarchical menu, the next menu could be shown and use the same method to select (c). As pointed out by [81], greater selection height and stable position of menu items are important to the usability of a hierarchical menu, so this design is promising as it expands selection height without changing the menu layout.

This method can be used for selecting targets packed in one dimension, such as single or hierarchical menu selection. For a hierarchical menu, as the hand is just located in the next menu after the selection (Figure 4-14.2(c)), so it only increases the movement distance for the last level. In addition to the rectangular menu layout and menu items, menu items and the expanding area could also be designed in other shapes, for example according to the interaction task or surface topology [13].

4.5.2.3 Targets Packed in 2D

Figure 4-14.3 illustrates selection for targets packed in 2D. There have been some efforts to address selection with occluded targets in 2D interfaces [182], while freehand 3D interaction introduces some novel opportunities. In this case, the expansion and reach selection cannot happen in the same 2D plane of the target, as shown in Figure 4-14.2, because all the targets are surrounded by others. However, the third dimension is available and can be used for expansion and selection. When the hand moves into a target, the target expands in another plane at a different depth along the Z dimension (a), e.g. 10 cm closer to the user's body from the current hand location. Then the user can move the hand in the third dimension to reach the expanded target area in order to select (b, c). Side view of the selection process is shown in (c). The blue line is the target and the orange line is the expanded target in another plane located at a different depth.

This method reveals the advantages of freehand and 3D interaction. In 2D interfaces, it is very difficult to expand the targets packed together in motor space to facilitate selection. In contrast, the combination of 3D interface and 3D freehand movement creates a novel opportunity for expanding the target and selecting in 3D motor space. This method can be used widely for interacting with existing 2D content using freehand gestures, such as selecting links packed in 2D web pages or icons packed on a 2D desktop.

4.5.2.4 Targets in Dense and Occluded 2D or 3D Environment

Figure 4-14.4 illustrates selection for very small targets or targets in a densely populated environment in 2D or 3D, with the target having a unique appearance. In this case, an area cursor (for 2D) or a cone (for 3D) are needed as the target is too small and/or overlapped with other objects. The basic process is similar to menu cone (Figure 4-2):

the user covers the target in the selection range (a), confirms the selection by a gesture (e.g. pull) to display the menu for disambiguation (b), and then disambiguates the target by reaching the target in the menu (c).

A challenge for such a method is that it cannot distinguish between the target and other objects with the same appearance [98]. This problem can be addressed by preserving the context of the first selection. Figure 4-14.5 illustrates a target selection method for such a case. It is a similar setting to Figure 4-14.4 but with all the objects having the same appearance. To select the target under this condition, the original target location and context has to be maintained rather than being rearranged into a menu. The basic process is also similar to Figure 4-14.4. The user covers the target (a) and then confirms the selection. Rather than arranging the targets into a menu, the whole selected area is expanded together to maintain the context (b). The user can then use the same method as in Figure 4-14.3 to select the target. If there is some occlusion between targets, a different viewpoint from another angle can be provided in (c) for better visibility.

Interacting with small targets using freehand interaction is challenging but often necessary. For example, there is a large amount of content already designed for desktop interaction. Items of this content are often small and densely packed, thus making them difficult to select with freehand interaction. As freehand gestural interaction becomes more common and pervasive, retrofitting suitable interaction techniques to such content will become increasingly important.

The point-expand-reach method has several benefits for interacting with such small and dense interactive content. The expanding target provides a large area for freehand selection, and visual feedback for the pointing action. Thus, users can adjust their input in the expanded area if they initially fail to make the desired selection. A larger selection area also makes the freehand Reach action easier. Overall, this method could make freehand selection easier and more accurate for small and densely packed targets. However, the design ideas presented here still require more extensive development, as well as further prototyping and evaluation.

4.6 Summary

Extending the option selection from Chapter 2, in this chapter we reported two studies about freehand gestural selection in dense populated environments to address the challenge of high precision interaction. The first study investigated target selection techniques in dense and occluded 3D environments, suggesting how we might design freehand 3D interfaces by extending lessons from experiences with body-attached or hand-held 3D tracking devices. And the findings suggest for freehand gestural selection, interaction design requiring a high accuracy single action may not be appropriate, while separating it into several connected low demand operations could be a possible solution.

The second study presented here is the design and evaluation of freehand gestural text entry methods. The designs proposed in this study, such as the circle layout, expanding target and the reach selection technique, are build upon the interaction techniques for desktop and touch surface. We combined these techniques together, and proposed the design for freehand gestural interaction. The virtual keyboard layout is familiar to Qwerty users and selection techniques are designed specifically for freehand gestural text input. These methods could be used for TV in living rooms or large interactive displays in public space. The user evaluations also demonstrated the effectiveness of our design.

Drawing on the results and findings from both studies, we also provided some design directions and example interaction techniques for effective and usable freehand gestural selection techniques for different tasks and scenarios.

CHAPTER 5

FREEHAND GESTURAL NAVIGATION FOR INTERACTIVE PUBLIC DISPLAYS

5.1 Introduction

The increasing use of large public displays in indoor and outdoor environment brings new requirements and opportunities for rich and engaging user experiences. There are many established means to interact with displays, such as keyboard, mouse and touch surface [102]. However, desktop input devices or touch surface are installed together with or close to the displays, so they are not suitable for many situations involving public displays. For example, these input devices are not suitable for interacting with displays in an enclosed space (e.g. screens in shop windows), or displays out of users' reach (e.g. displays in high locations), or large public displays that are intended to be viewed from a distance (Figure 5-1). Mobile devices such as mobile phones can also be used to interact with public displays, but configurations are normally needed to connect the personal devices to the public displays so again it is often not convenient or appropriate for users in public settings.

As we presented in Chapter 2, freehand gestural interaction, which has no need for hands-on input devices, is likely to become more important for interactive public displays. In addition, with improvements in the ability to visualize urban environments in 3D, some applications, such as virtual tours of a historic site or navigation around a planned plaza, can benefit from 3D user interfaces and a more immersive user experience. However, most established interaction techniques are in 2D, so they might be unsuitable for these scenarios. And most 3D gestural navigation techniques require a device mounted on the body, for example, holding a motion sensing device (e.g. Wii remote) or using fiducial markers that are tracked by a visual tracking system. Thus, we are motivated to design the interaction task of navigation, using freehand gestures

Figure 5-1: *Public display settings not suitable for desktop interaction techniques. Left: large display; Center, display in shop window; Right, display out of reach.*

tracked by only a remote sensor (e.g. Microsoft Kinect) without on-body attachments or devices.

In this chapter, we focus on the navigation task in 3D visualizations using freehand gestural interaction, as well as the investigation of freehand gesture in public space. We designed the freehand gestural navigation techniques and conducted two studies to evaluate them. A controlled experimental evaluation was performed to investigate navigation time and behaviour, comparing gestural navigation with a more traditional keyboard/mouse navigation technique in individual settings. We also conducted a more informal qualitative field study. We discuss our findings and suggest design lessons for freehand gestural navigation with interactive public displays.

5.2 Study 1: Controlled Quantitative Experiment

Our first study investigated freehand navigation with large displays in a controlled experiment. The gestural navigation was used to complete a searching task in a visualization of a 3D urban environment. To gain further insight into the characteristics of gestural navigation, we also compared it to a more conventional interaction method using keyboard and mouse.

5.2.1 Gestural Navigation Design

Given our aim to use low cost technology for freehand gestural interaction without holding or wearing a device or fiducial marker, it is difficult to track small movements of the finger or wrist (e.g. using a Kinect camera). Hence, the acceleration or movement of larger joints such as the hand or shoulder is used to control the navigation.

5.2.1.1 Movement

Previous navigation techniques with tracking devices provide some valuable references for freehand navigation design, for example, controlling the navigation speed and direction by relative motion using a single stick [51]. Moving the stick up and forward while pressing a button on the stick resulted in corresponding navigation movement. The navigation speed was specified by the stick's movement range while direction was controlled by rotation of the stick. In navigation techniques using a Wii remote [49], a method using a joystick for speed control, pitch for elevation control, and roll for direction control has better performance and user preference than another alternative navigation design.

Since with current low cost tracking cameras such as Kinect we cannot track the pitch and roll of the hand with high accuracy, navigation techniques designed for a Wii remote or similar hands-on devices are not suitable for freehand gestural navigation. Furthermore, compared to hand held devices, there is deliberately no joystick with freehand gestural interaction. So a navigation technique based solely on hand/body movement is needed.

A similar design was presented in [51], based on relative movement with a single 6DOF stick, in which the user can specify the initial position by pressing a button, setting a temporary and accurate initial position each time for calculating subsequent relative movement. However, as there is no button available for freehand gestural navigation, another method for setting the initial position is needed.

A candidate method is to use the same initial position for the entire navigation task, enlarging the initial position from a single point to a larger area to accommodate low accuracy freehand tracking. In this design, we set a sphere shaped initial position centered on the hand's starting position. When the user's hand reached out of this initial sphere, a vector was connected from the original hand position to the current position. The vector direction determined the navigation direction, and the vector length determined the speed of the navigation, i.e. the further the hand reached out, the faster the navigation, as shown in Figure 5-2(a). For example, users can move the hand right/left to strafe right/left, move the hand forwards/backwards to navigate forwards/backwards, and move the hand up/down to move up/down in the virtual 3D environment. With this design, only a single hand is needed to control the navigation movement, leaving the other hand available for other operations such as confirmation or pausing/restarting the navigation.

5.2.1.2 Looking Direction

For movement controlled by a single hand, the user's position in the virtual 3D environment changes but the direction in which the user is looking does not change. Although it is possible to use single hand position to control looking direction at the same time, i.e. when the hand moves left/right, the strafing and looking directions change synchronously, our pilot studies showed that it is difficult to navigate with this approach.

The rotation of hand or head has been used in previous studies [8, 49, 51] to change the direction in which the user is looking, however, these movements cannot currently be detected reliably with a single low cost camera. On the other hand, the rotation of large joints such as the shoulders can be detected effectively with a single camera. Given that shoulder turning is also involved in changing looking direction in the real world, we used turning of the user's shoulders to change the looking direction in our 3D system. When the user's shoulders turn to left/right, the looking direction turns left/right continually until the shoulders turn ahead again, as illustrated in Figure 5-2(b).

Thus, with a single hand to control the user's movement and the shoulder turning to control the user's looking direction, users can control navigation in the 3D environment by flying forward/back, strafing left/right, lifting the hand up/down, and turning their looking direction to the left/right. However, looking up/down is not available as an interaction technique technique, mainly because the body movement tracking by a single inexpensive camera is not sufficiently accurate or fast, so introducing pitch could make navigation difficult to control.

It is also possible to improve navigation performance by introducing different navigation techniques and settings according to environment context and scale, as suggested by [51, 99]. As this study is an initial investigation of freehand gestural navigation techniques, we focused mainly on the basic design without introducing advanced techniques for adapting to multi-scale environments or navigation context.

5.2.2 Experimental Evaluation

Locating places of interest is a common task in geographic applications or virtual tours in 3D visualizations. For example, the user can first find the rough location of a target in a satellite map or bird's-eye view, and then in order to see the target more clearly, the user needs to navigate close to the target, typically moving from a bird's-eye view to a street view. To gain more understanding of gestural navigation and its differences from established interaction techniques on such navigation tasks, we conducted a controlled experimental evaluation to investigate the effects of different target distances and directions.

5.2.2.1 3D Environment

We used CityEngine to generate the virtual cities that we used to investigate navigation. CityEngine is an L-system based procedural modelling system. It is capable of

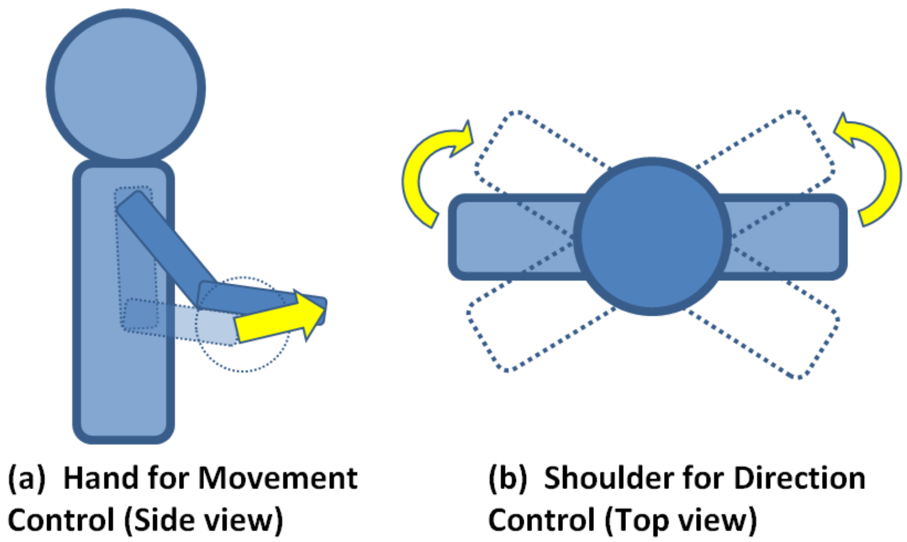


Figure 5-2: *Freehand gestural navigation design. Hand is used to control the movement (a) and shoulder is for direction (b).*

modelling a complete city using a comparatively small set of statistical and geographical input data and is highly controllable by the user.

We first designed the street network of our virtual city. We used a radial pattern to generate the streets in the city center, and an organic pattern for the city outskirts. This mimics the style of Paris's street network. Our city has 1619 streets in total. The street widths vary between 8 meters to 16 meters. The most complicated crossing is met by six streets. These streets also form "blocks" where buildings are located.

We then generated allotments from the street network by dividing the blocks into smaller units using a simple, recursive algorithm. Allotments were made convex and rectangular so they could be used as floor plans for generating buildings.

The buildings were generated in 18th century Paris style to create a realistic visualization. For example, the facades are decorated with windows and balconies and the roofs are mainly in the "zinc" and "ardoise" styles. The city contains 11507 buildings and is about $2000 \times 2000\ m$. The average height of all buildings is $20\ m$. The street widths varied between 8m and 16m.

5.2.2.2 Navigation Task

A cube, size $1m^3$, was used as the target. It was randomly coloured red, yellow or green and placed at the positions specified by TargetDistance and TargetDirection. Each target was placed 5m above the ground, and located in the center of a street which was 100m long and 10m wide, and whose direction was along a line from the start point to the corresponding target position, as shown in Figure 5-3. The start point

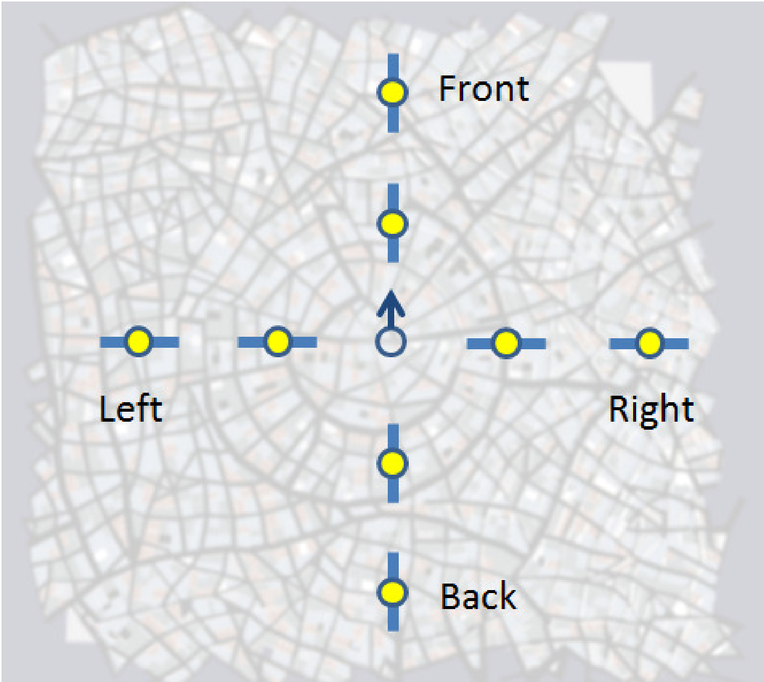


Figure 5-3: *The 3D city scale environment used in the experiment (Top view). The Blue circle in the center represents the start point and the arrow points to the initial direction. The yellow spheres represent the target and the blue lines represent streets containing the target.*

of the navigation had a bird's-eye view of the city. To require users to navigate down from the bird's-eye view to the street view, an occluder of size $100m^2$ was placed above the target. Consequently, users could not see the target unless they were close enough to the correct street (Figure 5-4). The users were required to find and report the colour of the target. A text description of the target location (Front, Back, Right, or Left) at the beginning of each trail, as well as a distinctive indicator (a yellow sphere of 10m diameter) at the same height as the start position, were used to help participants find the rough target location so they could concentrate on the navigation rather than on searching the target location.

5.2.2.3 Navigation Technique

For the conventional interaction technique, we used the "Freelook" navigation technique with a keyboard and mouse. Navigation movement was controlled by pressing the "W" key (forward), "S" key (back), "A" key (left) and "D" key (right) keys on the keyboard. The normal navigation speed was 37.5 m/s, and the user could use the "Shift" key to enter fast movement mode in which the navigation speed was 750 m/s, i.e. 20 times the standard speed The navigation direction is controlled by the mouse movement. The looking direction was controlled by mouse movement, i.e. turning the looking direction left/right or up/down by moving the mouse in the corresponding direction.

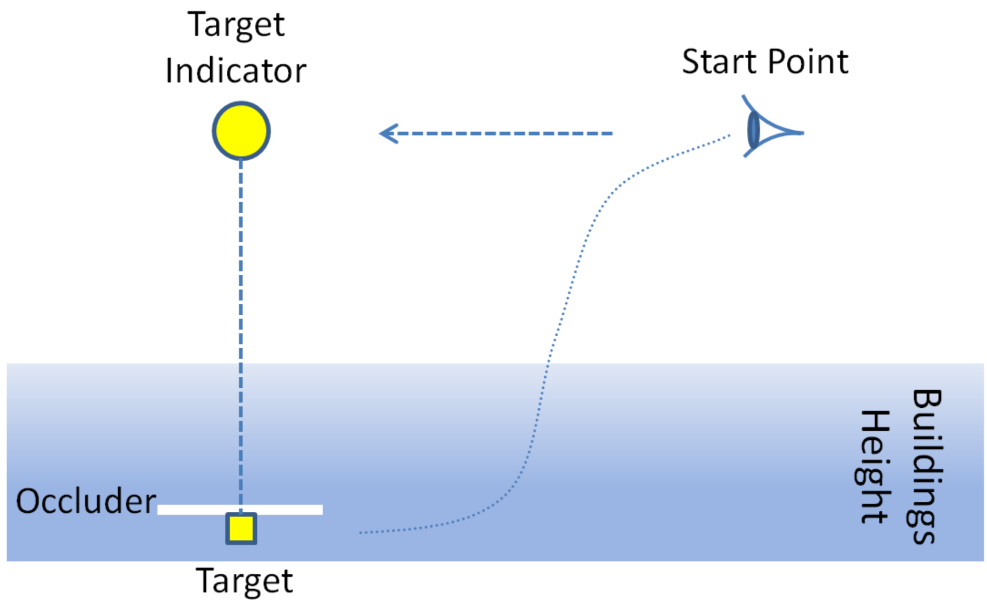


Figure 5-4: *The navigation task used in the experiment. The start point is at the height of a bird's eye view, and the user is required to navigate to the street view to see the Target (yellow cube) and report its colour. The Target Indicator (yellow sphere) is at the same height as the navigation start point, and the Occluder (white line) is placed to require the user to navigate down to street view height.*

For freehand gestural interaction, we used the navigation technique described in Section 5.2.1. A vector is calculated from the current hand position to the original position (i.e. user's hand position when the trial starts) to calculate navigation direction and speed, as shown in Equation 5.1 (d is the length of the vector in mm in real space, and V is the navigation speed in m/s in the digital space). Navigation direction changes continually when the user's shoulders turn in corresponding directions. The turn speed is two times the shoulder turning angle each second. The turn action is triggered only if the left and right shoulders' depth difference goes beyond 100 mm.

$$V = \begin{cases} 0 & (d <= 50) \\ 37.5 + 0.5(d-50) & (50 < d <= 200) \\ 112.5 + (d-50) & (200 < d <= 350) \\ 37.5 \times 20 & (350 < d) \end{cases} \tag{5.1}$$

5.2.2.4 Equipment and Setting

The 3D environment was displayed on a projection screen 203 cm in width and 115 cm in height. The height from the ground to the screen center was 145 cm. A Sanyo PDG-DWL2500 short-throw 3D projector was used with a 120 Hz refresh rate at 1280 × 720 resolution. Nvidia P854 3DVISION active glasses were used by participants to view in stereoscopic 3D. With keyboard/mouse navigation, the user sat in a chair of 45 cm height, placed 250 cm in front of the projection screen. The keyboard and mouse sat on a square table $80cm \times 80cm$ at a height of 70 cm. The mouse was placed on a mouse mat sized $23cm \times 30cm$. We used a Logitech K340 wireless keyboard (UK layout) and Logitech VX Nano wireless laser mouse. Users were free to place the table, keyboard and mouse in a comfortable position. With the gestural navigation technique, the user's skeleton movements were tracked using a Microsoft Kinect camera, with a refresh rate of 30 fps, and the OpenNI API on Windows 7. The Kinect camera was placed 50 cm in front of the projection screen at a height of 90 cm. The user stood 250 cm in front of the screen, so 200 cm in front of the Kinect camera. *OGRE* was used as the 3D engine.

5.2.2.5 Independent Variables

The independent variables used in our experiment are: navigation Technique (Keyboard/mouse, Gesture), TargetDistance (less than 400 m, greater than 800 m), and TargetDirection (Front, Back, Right, Left).

5.2.2.6 Hypotheses

- There will be a significant difference in navigation speed when participants use Keyboard/mouse vs Gesture.
- There will be a significant difference in navigation speed when participants navigate to target with different distances.
- There will be a significant difference in navigation speed when participants navigate to target with different directions.

5.2.2.7 Participants

We recruited 12 people (9 male and 3 female) ranging between 23 and 30 years of age with a mean age of 26.17 (sd = 2.41). All participants were right-handed. All participants were experienced computer users and had some experience of gestural interaction, such as using a Wii remote or Microsoft Kinect for games.

5.2.2.8 Procedure

For the keyboard/mouse technique, each trial started with a countdown and finished when the user pressed the space key to report the target colour. For the gestural technique, a left handed "hands up" gesture replaced the space key press.

5.2.2.9 Design

A repeated measures within-participants design was used. There were 2 sessions, one for each navigation Technique and their order was counterbalanced. In each session, there was a practice block followed by 2 test blocks; each block had 24 trials (3 trials for each TargetDistance and TargetDirection combination).

5.2.3 Results

Besides navigation time, we also recorded user behaviours, such as movement time in different directions and turn angles, for detailed analysis of the navigation task with the different navigation techniques.

5.2.3.1 Navigation Time

Navigation time was the time from the first navigation action (moving or turning) to the confirmation action (pressing the space key or raising the left hand). Some users accidentally triggered the confirmation before they saw the target colour, and 12 such trials (1.04% of the total data) were removed. A repeated-measures ANOVA for $Technique \times TargetDirection \times TargetDistance$ was used to analyse the results. Main effects were found for TargetDirection [1] ($F_{1.50,16.47} = 15.00, p < .001$), and TargetDistance ($F_{1,11} = 24.85, p < .001$). No significant effect was found for navigation Technique ($F_{1,11} = 1.06, p = .35$). No interaction effects were found.

Post hoc Bonferroni pairwise comparisons showed that Front TargetDirection was significantly faster than Right, Left and Back ($p < .01$). And the navigation time for the Short TargetDistance was significantly faster than for the Long TargetDistance ($p < .001$). Navigation time for the Short TargetDistance was significantly faster than for the Long TargetDistance ($p < .001$). Mean navigation time with keyboard/mouse was 7.85s ($sd = 5.25s$), and with gesture was 9.12s ($sd = 3.00s$). Mean navigation times for $Technique \times TargetDirection$ are shown in Figure 5-5, and for $Technique \times TargetDistance$ in Figure 5-6.

5.2.3.2 Navigation Behaviour

We also analysed the details of user navigation behaviours in terms of total movement distance and time, movement time in different directions, fast movement time, and turn angle to left/right with each navigation technique.

5.2.3.2.1 Total Movement Distance and Time A repeated-measures ANOVA for $Technique \times TargetDirection \times TargetDistance$ was used to analyze the navigation

[1] The sphericity assumption was not met so the Greenhouse-Geisser correction was applied; the corrected degrees of freedom are shown.

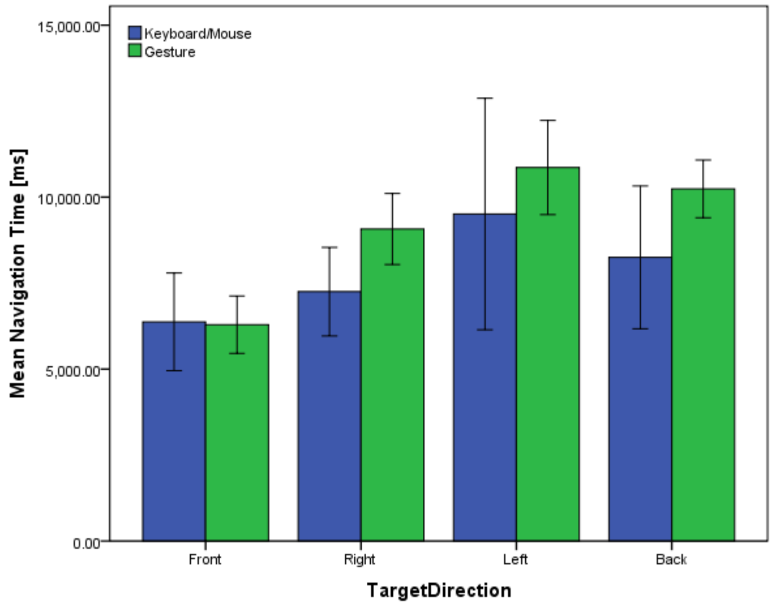


Figure 5-5: *Mean navigation time for navigation Technique × TargetDirection.*

distance. Main effects were found for TargetDirection ($F_{3,33} = 5.85, p < .01$), and TargetDistance ($F_{1,11} = 524.93, p < .001$). No significant effect was found for navigation Technique ($F_{1,11} = 3.68, p = .08$). No interaction effects were found.

Post hoc Bonferroni pairwise comparisons showed that Front TargetDirection had significantly less navigation distance than Right, Left and Back ($p < .01$). Navigation distance for the Short TargetDistance was significantly shorter than for the Long TargetDistance ($p < .001$). Mean total movement distance with keyboard/mouse navigation was 650.55m ($sd = 222.78m$), while with gestural navigation it was 696.18m ($sd = 249.42m$).

We also analysed movement time for the whole navigation process. Direction adjustment or target searching without moving were not counted in the movement time. A two-tailed dependent T-test found total movement time with keyboard/mouse navigation was significantly less than with gestural navigation ($t_{11} = -13.61, p < .001$). Total movement time with keyboard/mouse was 3.21s, while with gesture was 8.44s.

5.2.3.2.2 Movement in Different Directions We analysed movement time in different directions (forward, backward, leftward, rightward). With the keyboard, the user can move in only four directions and we recorded the time the user spent in each direction movement by pressing the “W” key (forward), “S” key (backward), “A” key (leftward) and “D” key (rightward). With gestural navigation, the user can control the movement more freely in every direction in 3D space. To compare with the results for keyboard/mouse, we separated the free gesture movement in 3D to four movement directions the same as in keyboard, as shown in Figure 5-7, and recorded the movement

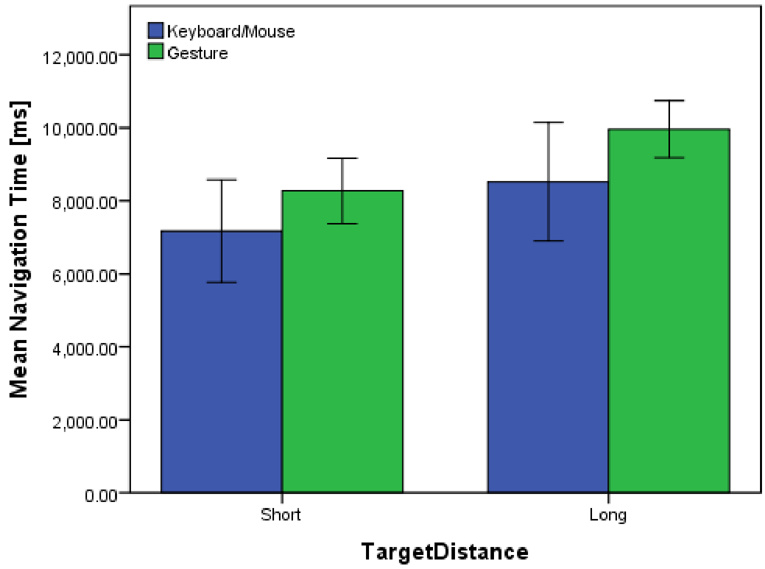


Figure 5-6: *Mean navigation time for navigation* $Technique \times TargetDistance$.

time towards each directions.

Analysis using a two-tailed dependent T-test found that forward movement was significantly more common with keyboard/mouse navigation than with gestural navigation ($t_{11} = 10.51, p < .001$). Correspondingly, movement in the other directions was less common with keyboard/mouse than with gestural navigation, i.e. backward ($t_{11} = -5.03, p < .001$), leftward ($t_{11} = -8.78, p < .001$), and rightward ($t_{11} = -4.08, p < .01$). The movement time percentages for different directions are shown in Figure 5-8.

5.2.3.2.3 Fast Movement Time With keyboard/mouse navigation, fast movement happened when users pressed the shift key. With gestural navigation, fast movement happened when the user's hand moved farther than 200 mm from the original hand position.

Analyses using two-tailed dependent T-tests found that there was significantly less overall fast movement time with keyboard/mouse than with gesture ($t_{11} = -3.47, p < .01$), but in contrast that the percentage of fast movement time over total movement time was significantly higher with keyboard/mouse than with gesture ($t_{11} = 2.89, p < .05$). Mean fast movement time for keyboard/mouse was 0.81s (26.63%) and for gesture was 1.49s (18.41%).

5.2.3.3 User Feedback

We collected user feedback by questionnaires after each session and the whole trial. Users were asked to rate from 1 (Strongly agree) to 7 (Strongly disagree): speed control (It is easy to control navigation speed), direction control (It is easy to control the

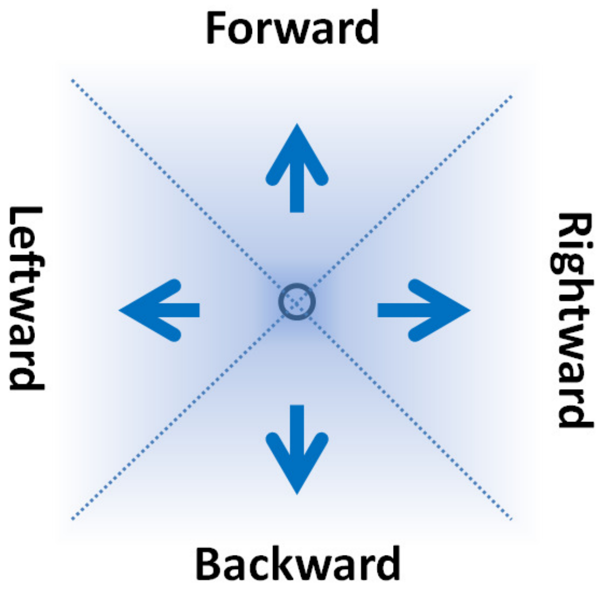


Figure 5-7: *Movement direction recording method for gestural navigation (Top view). The sphere in the center represents the initial hand position.*

navigation direction), comfort (I feel comfortable with this navigation technique) and preference (Overall, I like this navigation technique), as shown in Figure 5-9.

A two-tailed dependent T-test found that the keyboard/mouse technique was rated significantly easier to use than gesture for direction control ($t_{11} = -3.37, p < .01$). No significant difference was found for speed control, comfort and preference ($p > .05$).

A post-trial of NASA TLX test of task load index was used to measure perceived workload; the results are shown in Figure 5-10. Analysis using a two-tailed dependent T-test found no significant difference between keyboard/mouse and gesture for mental demand, physical demand, temporal demand, performance, effort or frustration ($p > .05$).

Participants also provided other comments. Two participants reported feeling uncomfortable when using the mouse and keyboard to navigate but much better with the gestural navigation. This may have been because when using the mouse and keyboard the visualization of the city moved following the mouse and keyboard, while using gesture the the visualization changed following the user's body movement. The latter is more similar to interaction in the real world. Participants also commented that the gesture navigation was more immersive and fun to use. Some participants said they preferred the speed control of gesture navigation because the speed increase was more graduated while for keyboard/mouse there are only two speeds to choose. Although the gestural direction control can feel more natural, participants said that using the mouse to control the direction control was more accurate and fast. This may, however, be due to the common massive greater experience using a mouse compared to gestural

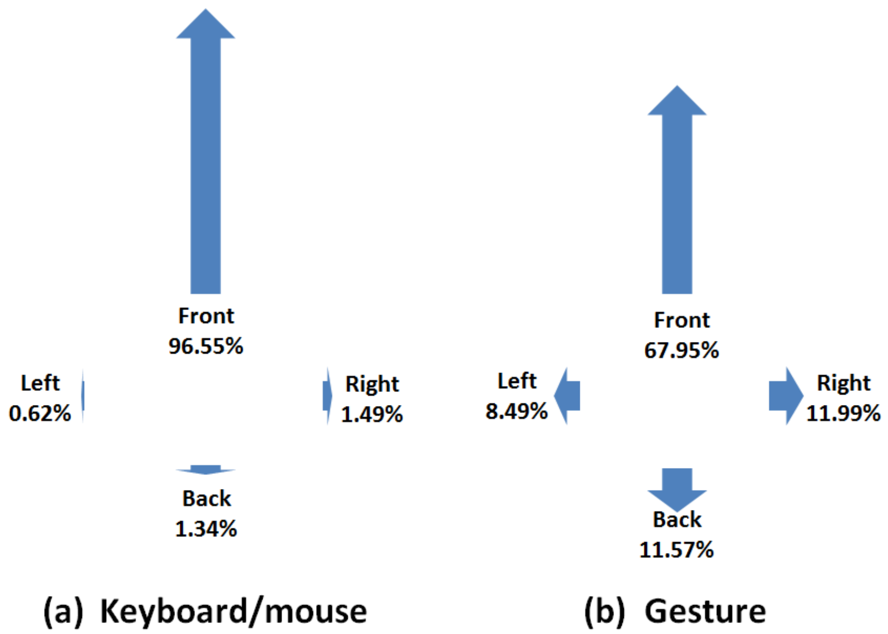


Figure 5-8: *Mean movement percentage in different direction Technique × Direction.*

interaction, which may change as gestural interaction moves beyond its infancy.

Participants also suggested some improvements to the gestural navigation design. Some participants said that although full 3D movement control using the hand has benefits, keeping the hand in the correct position to control navigation was a little tiring. It might be better to slow the speed in other directions to focus on forward movement as it is the primary movement direction. On direction control with the shoulders, some participants said it felt a little strange to turn their shoulders while keeping the head oriented to look at the screen. And it could be useful if head or gaze direction could be used to assist direction control. We noticed that some participants moved their head or body when they were close to the target in an attempt to see target. They mentioned this in feedback, expressing a desire for a way to control fine navigation adjustments with features such as head location and head direction.

With gestural interaction, many participants felt it was hard to stop the movement when they were near to the target or going in a wrong direction, while with the keyboard/mouse it was easy to stop by releasing the key. This could be addressed by design refinements suggested below.

Some participants also mentioned the remoter controller for a model tank which has two joysticks, each controlling one track of the tank. Pushing two joysticks forward together produces forward movement, and pulling both backward produces backward movement. Pulling one back and pushing the other forward produces a turn. The speed of movement or turn can be controlled by the joystick movement amplitude. This could

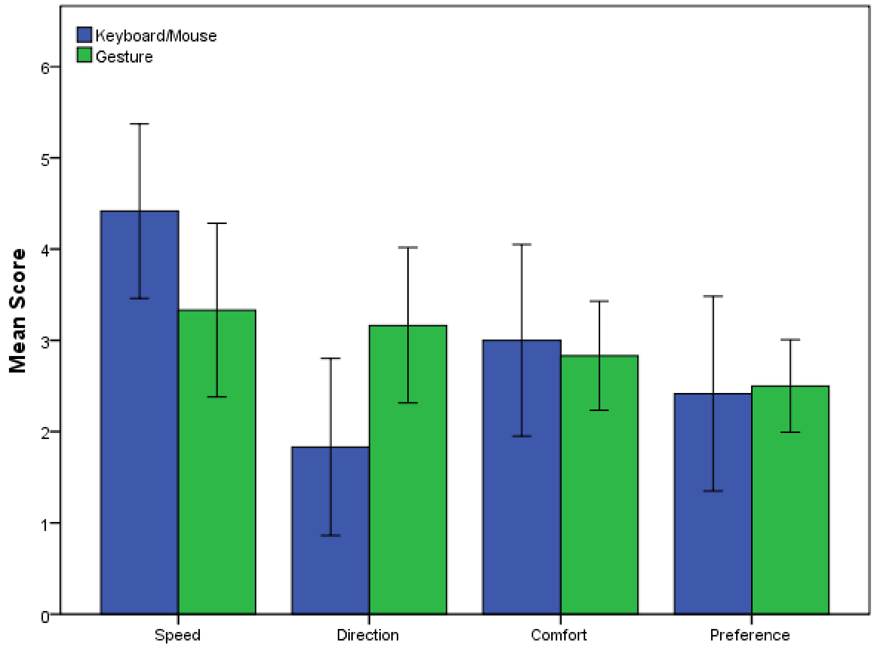


Figure 5-9: *User feedback about navigation speed control, direction control, comfort and preference.*

be a good metaphor for a double-handed gestural navigation control.

Participants also suggested that the context of the navigation environment could provide useful information and constraints, such as at street view height navigation being allowed only along a road, and turns allowed only at a junction.

5.2.4 Discussions

The results from study 1 show that while the gestural technique was consistently slower than using a mouse and keyboard, this difference was not significant. Given that the keyboard/mouse is often not practical to set up and use with interactive large displays in public space, the freehand navigation technique could be a promising method, offering walk up and use experience without any handheld or desktop input devices. For users who feel uncomfortable using a keyboard and mouse for 3D navigation, gestural interaction can also offer a possible way of reducing the discomfort.

The results indicate that gestural navigation shared some similar characteristics with the hands-on navigation technique, while there were also some interesting differences. For example, with freehand gestural navigation, users used more left, right and back movements, and the target direction had more effect on navigation time. This was influenced by the direction turn control with gestural interaction not being as easy and accurate as with a mouse, while participants exploited the greater freedom of 3D movement offered by freehand gesture to compensate. The findings reinforce our view that it is not straightforward for designers of freehand gestural interaction to

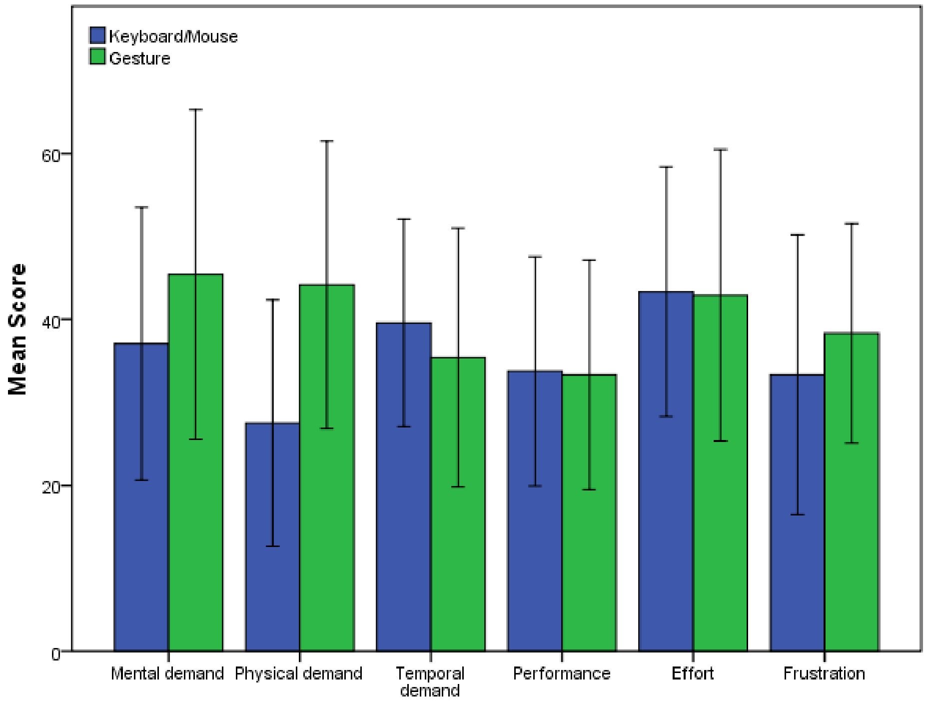


Figure 5-10: *NASA TLX task load index results.*

adapt their experience of traditional hands-on input devices to the design of freehand gestural interaction.

5.3 Study 2: Qualitative Field Study

In the previous lab based study, we found similar performance with gestural and mouse/keyboard navigation, with some differences in navigation behaviour. The experience of gestural navigation was described by our participants as more fun and natural, and these are indeed important characteristics in many applications of interactive public displays. However, there are always severe constraints on how much fun and how natural any experience can be in the controlled setting of a lab-based experiment. To gain further insights into these aspects of our interaction techniques, we conducted a much more informal field study in which primary school children used both keyboard and mouse and hands-free gestural interaction to play a game on an interactive public display.

5.3.1 Design

Based on the results and feedback from the first study, some modifications of the freehand gestural interface design were made. We used two hands to control both movement and direction turn, as direction turn may be easier and more natural to control with

(a) Broomstick Navigation Concept

(b) Navigation Control (Top View)

Figure 5-11: *Broomstick navigation design*

the hands than with the shoulders. To make the controls easy to understand, we used a flying broomstick as the metaphor, and this was found to be a familiar concept for 3D navigation from books and films enjoyed by our participants.

To make the flying broomstick easier to control, we added a "handlebar" towards the front of the broomstick (Figure 5-11a). Users put their hands parallel and close to their waist to start. After the broomstick starts flying, the center point of the hands is used to control the movement in the same way as in previous lab study. Turn the looking direction is similar to cycling: the more users twist their hands and move towards one side, the faster the looking direction turns. The degrees of freedom with the broomstick navigation control are the same as with the gestural navigation technique in previous lab study.

5.3.2 Field Evaluation

We conducted a qualitative evaluation with primary school students. We designed a game in which users operate a flying broomstick to navigate in 3D space, hitting targets and avoiding obstacles. The students played this game in groups in a classroom, using large interactive displays.

5.3.2.1 Navigation Task

The game involved targets (ogres) and obstacles (penguins). Users needed to hit 5 ogres to successfully finish the task, and if they hit 2 penguins, the task was failed. There were 54 targets and obstacles placed in a 3D space 180m x 180m x 360m. This space was evenly divided into 54 cubes 60m x 60m x 60m, each with one target or obstacle placed at a random location inside. The target/obstacle type was allocated randomly with a 1/5 chance of it being a penguin and 4/5 of it being an ogre. The navigation start point was 180m in front of this space.

5.3.2.2 Equipment and Setting

The equipment and setting were almost the same as for the lab-based study as described in Section 5.2.2.4, apart from the equipment being placed in a primary school classroom rather than in a laboratory. And OpenNI was replaced by the Kinect for Windows SDK to make switching between participants in the same group easier, without the need for a calibration pose. To ensure that every participant in the group could view the 3D navigation at the same time, the stereoscopic 3D effect was turned off so everyone could view the scene without stereoscopic 3D glasses.

5.3.2.3 Participants

We recruited 46 primary school students (24 male and 22 female) ranging between 6 and 10 years of age. All participants had some experience of gestural interaction, such as using a Wii remote or Microsoft Kinect for games.

5.3.2.4 Procedure

The participants performed the experiments in groups. The experimenter demonstrated the navigation technique and the task to the participants, and then each group of participants took turns to perform the 2 sessions with different navigation techniques. Each session lasted 10 minutes (largely determined by the school timetable), and the participants kept performing the task until their 10 minutes was up. The number of participants in each group was randomly assigned as 4 or 5 by the teacher in the school depending on the children's timetables before their trials. We randomized the order of navigation technique for each group as we could not control the number of participants in each group and the exact total number of participants. Participants were gathered in the room about 2 minutes before their trials, and were engaged in their normal activities before that.

5.3.3 Results

We recorded the navigation time of successful trials, the percentage of successful trials, and asked the participants for their feedback about the interaction techniques. As the number of participants in each group and the number of trials per participant with each technique could not be controlled in this field study, we report only descriptive data rather than analytical statistics. For gestural navigation, the average navigation time was 24.61s (sd = 12.34s), and the success rate was 80.90% (72 out of 89 trials).

For keyboard/mouse input, the average navigation time was 26.94s (sd = 12.33s) and the success rate was 93.16% (109 out of 117 trials).

The participants were also asked which of the two techniques they preferred, having experienced both. 24 participants preferred the gestural navigation and 22 preferred the keyboard/mouse.

Some participants said that with the gestural navigation method they could control the navigation in a more natural way with body movement, e.g. leaning forward in order to put their hand further forward to move faster, and turning to control direction. They also commented that gestural navigation was "cool" and they had a lot of fun using it. Some participants thought the mouse and keyboard was easier to use because they were very experienced using them to interact with standard computers.

5.3.4 Discussions

5.3.4.1 User Performance and Preferences

Navigation time and user preferences were very similar between the two techniques, which accords with the first study. The participants finished more trials in total with the keyboard/mouse mainly because it took longer to switch navigation control between users with gestural navigation, e.g. the current player needs to make sure she is being tracked by the Kinect camera before each trial, and it often took some time to get used to starting the navigation with gestural interaction.

Participants completed more successful trials with keyboard/mouse than with gesture. Users have tactile feedback when pressing physical buttons and so can control the navigation state (moving/stop) more effectively. With freehand gestural navigation, all control actions are performed "in the air" without tactile feedback and navigation state control is not so easy, leading to more errors.

A possible solution is to provide more feedback (e.g. visual or auditory) of the user's current body position, as well as the navigation state of the broomstick, so users can have a better perception of their subjective body movements, as well as the state of the artefact they are controlling.

5.3.4.2 Group Behaviour and Experience Sharing

We also observed that with gestural navigation the group's behaviour was more interactive with each other. The group members had more conversations, and the non-playing group members engaged more in the game and shared the experience with the current player. For example, when the current player navigated towards a penguin, other members would shout to alert her, and tried to guide the player by speaking directions or even moving their own bodies as if they were controlling the navigation themselves. The current player also tried to perform according to other group members' contributions. In contrast, with the keyboard/mouse interaction, the participants had less communication and typically the current player simply finished the game in her turn, quietly concentrating on the display, keyboard and mouse.

This is mainly because that with gestural navigation technique, the group members could observe and understand the player's actions immediately from their hand/body movements, so they could share the player's experience more directly. With keyboard/mouse, it is difficult for others to distinguish between the user's key presses, so less experience is shared in the social group.

This observation suggests that freehand gestural interaction provides a more "natural" user experience not only from the perspective of navigation control by a single user, but also from the perspective of sharing the user experience within a social group in a public space. Our observation that players adapt their gestural interaction according to the behaviour of other people in the same social space aligns with Hinrichs and Carpendale's [80] finding that gestures were influenced by the social context. Furthermore, not only is gestural interaction affected by the social environment, we also find that gestural interaction can enhance the experience of the social environment compared to conventional desktop interaction techniques.

5.4 Design Suggestions

In a pair of complementary studies, we found performance and preference with gestural navigation to be similar to performance with keyboard/mouse. Considering that desktop input devices are often difficult to set up and use in the setting of a public space and large interactive display, gestural navigation techniques enabled by an inexpensive single camera are a very promising interaction technique for interactive public displays.

Furthermore, gestural interaction can deliver a more natural and immersive experience of navigation tasks. The fun experience of gestural navigation was one common response from our participants. This suggests that gestural interaction can be a suitable technique for interactive displays used for exhibitions, planning or commerce, e.g. using gestures to navigate a virtual reconstruction of a historical site, to explore and evaluate a planning building/area, or to explore products in 3D.

When interacting with displays in public settings, gestural interaction can enhance not only the experience of the current user but also the social experience of the group of people sharing the same public space or activity. The current user's activity, behaviour and even emotion can be shared in a more effective way than with desktop interaction techniques, as bystanders can observe, understand and appreciate the player's experience better thanks to the natural interaction of freehand gesture. This suggests that freehand gesture should be explored further as an interaction technique for shared interactive public displays.

Despite the potential benefits of freehand gestural interaction, there are also challenges for designers of interactive public displays using gestural interaction. For example, in both studies we noticed that some actions (e.g. stop, turn) were more difficult to perform with gestural navigation than with keyboard/mouse. Possible solutions include providing more visual or auditory feedback, or using additional input from other modalities, e.g. speech. However, not all combinations will be effective in settings where there is considerable visual or auditory noise or where some modalities may be inappropriate. Unaccompanied freehand gestural input remains potentially useful and effective in many scenarios with interactive public displays and further work may show how far we can push its use and effectiveness.

5.5 Summary

In this chapter, we reported a set of two studies using freehand gesture to control the 3D navigations in lab settings and in public environments. It is also an investigation about the opportunities of freehand gestural interaction for interactive public display and 3D interaction, The experimental evaluation recorded a similar performance from freehand gestural technique comparing to desktop navigation devices, while with a more natural experience. More importantly, the findings indicate that freehand gestural interaction can enhance the experience sharing in public settings thus improve the interactivity for interactive public displays.

CHAPTER 6

SUMMARY AND FUTURE WORK

6.1 Summary

This research is an attempt to explore the effective design of freehand gestural interaction. The fluidity and immediacy of multi-touch interactive products suggests that freehand gestural user interaction could be a potentially fruitful direction for interaction design, especially for scenarios not suitable for touch screen or desktop interaction techniques, such as interacting with large displays in public space or TVs in living rooms. The design guidelines and issues for freehand gestural interaction are still largely unexplored and thus motivate this research (Chapter 1).

We first provided a review of related works in the area of gestural interaction (Chapter 2), including discussions of taxonomy and usability of gestural interaction, various enabling techniques of gestural interfaces (e.g. touch surface and hand-held motion tracking devices), as well as gestural interaction design for different tasks (e.g. selection, text entry and navigation). Based on the previous work, we analysed the characteristics and design space of freehand gestural interaction by comparing the similarities and differences of freehand interaction and existing gestural techniques. We also summarized two design challenges of **gesture delimiters** and **high precision interaction**, as well as opportunities of **interactive public display and 3D interaction** with freehand gestural interaction. All the analysis builds the ground for the studies in the following chapters.

We then progressively investigated several aspects of freehand gestural interaction about its design challenges and opportunities. In Chapter 3, we focused on addressing the challenge of **gesture delimiters**, and investigated a basic and important user interaction task of option selection. We presented two related studies in this chapter. In the first study, we proposed two freehand gesture selection techniques, stroke and reach, based on the previous research and design on selection techniques for touch surface and desktop. And we evaluated the proposed design in a user evaluation, the results indicate that Reach selection technique has better user performance and preference thus has the potential to be widely used for freehand gestural interaction. And in the second study, we extended the reach technique to 3D interfaces by designing different menu 3D layout, and the investigation results show strong directional effects, which could influences performance and usability.

Then in Chapter 4, which builds upon the findings in Chapter 3, we investigated the selection task in densely populated environments to address the challenge of **high precision interaction** with freehand gesture. Again, we presented two related studies in this chapter. In the first study, we reported the design and evaluation of the selection task in a dense 3D environment. We found that interaction design requiring a high accuracy single action are not appropriate for freehand gestural selection, while separating it into several connected low demand operations could be a potential solution. The second study is about text entry with virtual keyboard, which is essentially a series selection of characters from a densely tiled keyboard. We presented 3 selection methods and 2 keyboard layout, which based on the option selection studies in chapter 2 and the QWERTY keyboard layout. The user evaluation shows reach and expand selection method and circle layout could enhance users' performance and reduce the workload.

Following Chapter 3 and 4, which focus on addressing the design challenges, we investigated the opportunities of freehand gestural interaction for **interactive public display and 3D interaction** in Chapter 5. In this chapter, we designed 3D navigation techniques and evaluated in both lab and public settings. Based on the interaction design, prototype development and user evaluations, we gave the results of user performance, behaviour and preference. We compared the freehand gestural interaction with traditional desktop interaction (i.e. keyboard and mouse), and found performance and preference with gestural navigation to be similar to performance with keyboard/mouse. Furthermore, freehand gestural interaction can deliver a more natural, immersive experience of navigation tasks. Gestural interaction can also enhance the group experience and collaboration for users sharing the same public space or activity.

Finally, we summarized the thesis in this chapter (Chapter 6). In the following sections, we provided a set of practical design suggestions for effective freehand gestural interaction design for different scenarios and interaction tasks, as well as the future work and conclusions.

6.2 Design Suggestions

Based on the studies and findings in previous chapters, here we summarize some overall practical design suggestions for freehand gestural interaction design:

- Hand fatigue
  - It is not appropriate to require users to perform single actions with high accuracy, or to keep their hands in the air for a long time. This is mainly because the hand is moving without any physical support, so these actions will make the arm fatigue more quickly. One possible solution could be to separate a highly demanding interaction task into several connected low demand tasks. This could not only reduce the physical demands, but also potentially achieve better performance. Furthermore, it is also useful to provide a means to rest the user's arm if the task requires some time to finish.
  - Frequent movements in the Z dimension are not recommended for freehand gestural interaction. This is not only because the hand moves more slowly forward and backward [68], and actions in the Z dimension can be error-prone [142], but also because frequent movements in the Z dimension can increase the hand movement distance and corresponding physical demands.
- Selection design
  - The Reach technique, building on the goal crossing technique [2] in 2D interfaces, could be a useful method for target selection with 3D freehand gesture, as it does not require any confirmation trigger for selection. Although finger or hand poses could be used as a trigger, more precise tracking equipment would be required (e.g. data gloves [126]), or the tracking could be sensitive to viewpoint changes [157, 71] when using a single camera. With the Reach technique, on the other hand, users can select the target using their experience of reaching for an object in real life, without the need for a finger or hand pose as a gestural confirmation or delimiter.
  - Combining the expanding target and reach selection techniques is useful for freehand gestural interaction, especially for hand motion tracked by inexpensive sensors with low resolution. When used with the reach technique, the target can expand not only in virtual space but also in motor space, which addresses the limitation that targets can expand only in virtual space [122]. It is also well suited to freehand interaction techniques due to its error tolerance.
  - The circle layout is useful for interface design with freehand gesture. In the circle layout, all items have one side facing to the centre and, therefore, can be reached directly without moving the hands over other objects. And the large blank area in the centre can allow users to relax their hands without

worrying about triggering undesired actions, thus potentially reducing arm fatigue.

- Dimensions and directions
  - Even when working in 3D, freehand gestural interaction designers should consider mapping 3D hand movements to 2D interaction for simple user interfaces with few elements. The number of targets located forward or backward should be limited as the hand moves more slowly forwards and backwards [68], and can be error-prone. Designers should also think carefully before committing to a full 3D interaction task, because such a task can be more demanding than a similar task in 2D and could be slower and more error-prone for freehand 3D interaction.
  - One way to use hand motion in the Z dimension is as the delimiter or trigger to invoke certain commands or to make confirmations. For example, the pull gesture is used in the menu cone to connect between gestures (Figure 4-2). Given that a finger cannot currently be tracked reliably as a gesture delimiter by a single low-cost camera, and hand movements in the X-Y dimensions can be used for faster and more accurate actions, using hand movements in the Z dimension as the gesture delimiter is a potentially useful addition to the designer's toolkit.
  - Keeping the user's hand movements within the range of their comfort zone and avoiding uncomfortable areas or directions is important for freehand gestural interaction design. As in both 2D and 3D selection, we found that users have preferences for different hand movement directions, e.g. they do not like cross-body movements such as left-down or left-back.
- Freehand gestures in public space
  - When interacting with displays in public settings, gestural interaction can enhance not only the experience of the active user, but also the social experience of the group of people sharing the same public space or activity. The current user's activity, behaviour and even emotion can be shared in a more effective way than with desktop interaction techniques, as bystanders can observe, understand and appreciate the player's experience better thanks to the natural interaction of freehand gesture.
- Visual feedback
  - Providing continuously visual feedback will enhance the usability and users' confidence when they perform the interaction, especially for the interaction

techniques using hands location or gesture of direct manipulation. The visual feedback could give information and hints not only to the current users, but also to other audiences in the same public space, about the current state of the interaction.

6.3 Future Work

In an attempt to provide some initial steps towards effective freehand gestural design, it is difficult to exhaust all research topics in this area. In this section we list some future research directions of freehand gestural interaction.

6.3.1 Input and Output of Freehand Gestural Interaction

Many existing devices could serve as input or output for freehand gestural interaction. There are many emerging techniques and sensors under development currently. Here we analyse some possible input and output techniques for freehand gestural interaction.

6.3.1.1 Gestural Input and Multimodal Interaction

In all the studies presented in this thesis we used Microsoft Kinect as the tracking sensors. This is mainly because our research aims to facilitate freehand gestural interaction tracked by a low-cost remote single camera with no requirement for a user to carry an input device or wear fiducial markers. Microsoft Kinect is a typical low-cost remote gesture tracking sensor available when the studies were conducted. Some more accurate tracking systems, such as Vicon cameras[1], can offer high accuracy but they require special markers attached on the body and calibration before use, which we explicitly want to avoid in freehand gestural interaction.

One obvious drawback of Microsoft Kinect is the lack of accurate wrist and finger tracking due to its low camera resolution. Although the newer version offers better tracking resolution, it is still not enough in its effective tracking distance. Other sensors, such as the recent publicly released LeapMotion[2] sensor, could provide very accurate wrist and finger tracking without markers. The tracking range is relatively short compared with the Microsoft Kinect, while it is difficult to track the whole arm and body.

Wearable devices may offer another opportunity for freehand gestural interaction. Compared to gesture tracking enabled by fiducial markers, wearable devices such as a ring, watch, wristband or cloth are more user friendly as people would like to wear and use them in everyday life. Thus wearable devices may apply widely as gestural input devices. And as they are attached to the human body, it is easier to detect different

[1] http://www.vicon.com/
[2] https://www.leapmotion.com/

information with different sensors, such as wrist/finger posture, motion and electrical activity in the muscles.

Another possibility to improve freehand gestural interaction is multi-modal interaction. Multi-modal systems represent an interaction paradigm shift away from conventional windows-icons-menus-pointers (WIMP) interfaces to more natural flexibility and portability by providing users with greater expressive power and transparent experience [131, 132]. To achieve this the multi-modal interaction systems should integrate complementary modalities. So the strengths of each mode are used to overcome weaknesses in the other. Considering that humans always communicate with each other in multiple modalities, such as speech, gesture and gaze, multi-modal systems can potentially effect a more natural user interaction.

There are some benefits to using different parts of the body for gestural control. For example, in real life, users normally control fine movements with fingers while using arm or body movements for less accurate tasks in a larger range. If we can combine finger, hand and body movement together, the user interaction could be more natural and effective. Other usages of multi-modal interaction include speech [23], eye gaze [161] and body sensing [167, 75]. Processing different interaction modalities in parallel with gestures for interaction could enhance the user's preference and performance.

6.3.1.2 Tactile and Visual Output of Freehand Gestural Interaction

One main challenge of the freehand gestural interaction is the lack of the tactile feedback. With traditional desktop input devices, such as the mouse and keyboard, the keys and buttons could provide tactile feedback so that users have clear feeling of the device movement or button press/release/hold with small movements of the finger or hand. It is also the case for 3D hand held tracking devices mounted with buttons, such as Wii remote or Sony Move.

Tactile feedback could bring many benefits proved in previous research. For example, for wearable devices such as the data glove, finger-based tactile feedback could help users to perform direct manipulation tasks with more reliability [149]. And for hand-held devices, the tactile feedback could significantly improve the performance. For example, to perform the laparoscopic task in a virtual reality environment, the tactile feedback could significantly enhance the reaction time [177] than with visual feedback alone. For a mobile collaboration environment, the tactile feedback could also enhance the experience. Tactile feedback can reduce the overload of information in visual space and gently guides the user's attention to an area of interest [190].

However, for freehand gesture tracked by remote sensors, such as Kinect or Leap Motion, one significant difference is the lack of tactile feedback: hands only move in the air without any tactile feedback. For some interaction tasks, such as selection or direct manipulation, it is not very "natural" as the touch and feeling of the objects could be essential for a better understanding and performance in such scenarios.

There are several new input and output techniques which could be used to introduce the tactile feedback in freehand gestural interaction. For example, very thin glove devices could provide tactile feedback on finger tips with semiconductor nano materials [193], which could be used in applications such as enhancing surgical and operation gloves. In other scenarios where users cannot wear special gloves, the ultrasound could be another possible solution [3, 82]. And with the development of more expressive tactile devices [3], the wearable devices, such as watches, rings and wristbands could also be used as methods to introduce tactile feedback for freehand gestural interaction.

Another fast development direction is towards different types of displays. These displays require different techniques for interaction from the traditional desktop interaction techniques, and freehand gestural technique could be a potential method. For example, head mounted displays like Google Glass[4] are becoming popular now, and when users want to interact with virtual or real objects through their head mounted display, pointing their hand directly to the object in the air could be a very straightforward and natural way. With non-flat displays, such as a curved large display [5] and 3D volume displays [63], freehand gestural interaction could also be more suitable than 2D input methods for the interactive content presented in 3D space.

6.3.2 Applications of Freehand Gestural Interaction

Many freehand gesture tracking devices were originally designed for gaming. Thus currently entertainment is still one of the main application areas of freehand gesture interaction. However, with the development of motion sensing techniques, data analysis and visualization, as well as connectivity among devices, more applications will emerge.

6.3.2.1 Interaction in Living Room and Public Space

Internet connectivity is driving a rapid increase in the range and scope of interactive experiences on the TV platform, and it represents an exciting new opportunity for developing new practice and methodology for the living room experience. For example, more interactive content is now available for users' consumption. However, the traditional TV remote is limited in function and thus not suitable for such interactivity. On the other hand, some products focus on living room experience, such as Xbox One [6] and Samsung Smart TV [7], provide a freehand gestural interaction technique. Users can navigate the menu system, control the image and video playback or play games. Although current interaction technique is still in the early stage of development and

[3] http://immersion.com/
[4] http://www.google.com/glass/
[5] http://www.lg.com/us/oled/ultimate-design.jsp
[6] www.xbox.com/xboxone
[7] http://www.samsung.com/us/2013-smart-tv

with much space for improvement, these products still prove that freehand gestural interaction now goes beyond research labs to consumer markets.

Furthermore, living room experience is not limited to a single TV display currently. Many devices, especially mobile phones and tablets are now used in the living room very often when the TV is on. Furthermore, various web sites and services are offering a variety of experiences such as personal broadcast, video on demand or social networking. Facilitating discovery of devices and services, enabling synchronization of content among those to ensure seamless experiences, are also challenging in the living room. In everyday life, people are very good at using hand movements and gestures for various tasks such as communicating with people, organizing documents on a desktop or assembling a toy. Drawing from such natural user experience, freehand gesture could be a useful technique for interaction with heterogeneous devices and services.

Similarly, the digital content in public spaces, e.g. meeting rooms, classrooms or shopping mall, is also expected to be more interactive. Such interaction scenarios are not limited to a single user, but extend into collaborative environments with multiple active users and potential audiences. Freehand gestural interaction which is naturally used for interpersonal communication, can be easily understood among collaborator and audiences, so can potentially improve the effectiveness of interaction in public space.

6.3.2.2 Information Visualization and 3D Graphics

Currently large amounts of data of various types are available for analysis. They come from various sources and increase rapidly every second, such as from on-line services like social networks, search engines, business support, data from sensors mounted on mobile devices or in the environments. To build a deeper understanding of such large amounts of data for revealing the underlying truth, information visualization is essential to provide proper data representation and interaction for users.

Data representation, which largely relates to computer graphics, has received the vast majority of attention in previous research. The interaction with the data visualization has not yet to take full advantage of these new possibilities in interaction technologies, as they still largely employ the traditional desktop, mouse, and keyboard set-up [110]. Considering that data visualization is moving into new contexts, such as public space (e.g. museums and meeting rooms), hand-held devices (e.g. mobile phones and tablets) and wearable devices (smart watches and head mounted displays), it would benefit more ordinary people by providing data access and interaction in pervasive settings. Freehand gestural interaction, which could provide such “walk in and use” user experience, could also be a very potential candidate for information visualization.

The creation of visualizations and the exploration of data have a number of dedicated underlying interaction requirements provide challenges that visualization research is just beginning to address. Large data visualization can achieve a high-level of visual complexity quickly, thus effective interaction techniques for information visualization are essential. Yi et.al [192] summarized the interaction techniques as seven general categories: 1) Select, 2) Explore, 3) Reconfigure, 4) Encode, 5) Abstract/Elaborate, 6) Filter, and 7) Connect. How to design the effective and natural interactions for these operations in different applications and scenarios is still challenging.

For example, interaction in augmented reality (AR) environments containing a mixture of virtual and real objects can be a challenging task. Google Goggles [8] or Google Glass can use a camera on a mobile phone or wearable glasses to help search for an object in view, or show the user's location and direction in a 3D environment. However, AR inevitably brings some density and occlusion issues to user interaction with overloaded virtual information, thus proper interaction design for tasks mentioned in [192], such as selection, navigation and filter is essential for AR applications.

Similar to information visualization, the use of 3D graphics is also increasingly important in more areas of application from home entertainment to engineering and medical systems, and many others. More and more information and other content is visualized and manipulated in 3D, bringing a corresponding increase in the importance of effective and usable 3D user interfaces. However, the mapping of 3D tasks to 2D presentation and interaction may not be particularly natural, and consequently are less effective and usable for interacting in 3D. 3D user interfaces consisting of special tracking devices are rarely adopted by ordinary users. Thus, freehand gestural interaction could also bring a promising future for 3D user interfaces.

For example, the film industry currently relies largely on the 3D animation and effects in the film. However, it is always complicated to achieve effective communication between the director and the effect artist resulting in a high cost of rework. If the target effect cannot only be described, but can also be created roughly by the director's freehand gestures in 3D without the requirement of special devices and learning, it could be more effective and reduce both the time and the financial cost. Similar 3D interaction could also be used for education or medical services, e.g. viewing and manipulating chemistry or biological structure in 3D, for meetingrooms/classrooms or on-line settings.

6.4 Conclusion

People use freehand gestures quite often in daily communications, so it is considered as a form of natural user interaction method [20], which could leverage users common knowledge of everyday life without special training or learning. Although gestural interaction has been investigated for many years, most previous research uses hand

[8] `http://www.google.co.uk/mobile/goggles/`

held devices or wearing fiducial markers for gesture tracking. The freehand gestures, which are tracked by distance sensors without requiring users to hold or wear special devices, are not fully explored in previous research, even though they could be easier to use and deployed for ordinary users in every day life. There is very little knowledge and understanding about the human factors and interaction design issues with freehand gestural interaction, which may lead to less effective interaction design and affect its usage.

At the same time, users increasingly expect more interactive experiences for applications including learning, gaming, urban visualization and planning. And with gestural input becoming increasingly popular and moving beyond home gaming settings, freehand gestural interaction, which enables easier and more convenient "walk up and use" [13], is likely to become more important in various everyday settings. Furthermore, applying freehand gestural interaction together with some other fast developing techniques, such as information visualization and 3D user interfaces, could offer rich, natural and immersive user experiences to ordinary users with direct and easy interaction. Thus it is important to gain a better understanding of freehand gestural interaction.

In this thesis, we presented a series of design and studies leading to a better understanding of effective freehand gestural interaction design. We analysed the design considerations of freehand gestural interaction, focussed on its unique characteristics and differences with other well known studies of interaction methods. Building upon such analysis and understanding, we then conducted several evaluations for several different interface designs. While reporting the design considerations, user interfaces prototypes, experimental design and results, we also provide many practical suggestions for freehand gestural interaction designers based on our observations and discussions. As this thesis is an early work of its type, it is not possible to exhaust all the different interaction tasks and relevant topics. In the above sections we presented some future work directions, which we believe are interesting work which could be continued by ourselves or others. The research in this thesis focuses mainly on some basic interaction issues and human factors, and demonstrates how to design freehand gestural user interfaces with several examples. This research can serve as ground work for the future development of freehand gestural interactions.

APPENDIX A

EXPERIMENT DOCUMENTS FOR STUDIES

Participant No...............

Gesture Selection Study

Thank you for taking part in this study! The aim of the study is to gain an insight into the usability and performance aspects of gestural interaction in 3D selection task. The study will be conducted for near one hour. And contain four sessions and one questionnaire in the end. Each session has a training block and a test block.You can withdraw at any time before, during or after the study.

Data Collection For This Study: During the study we will be filming you. This video may be used in presentations. Anonymisation of video is not easy, so, if you are willing for your video to be shown in presentations then please indicate below.

Additional Data Collection: In addition to being filmed we will be recording data from a Kinect camera. We are collecting this data for study the patterns of body movement during gestures.

Handling of Data: All data will as far as possible be anonymised – video data can only be identified by your image. All data will be stored securely.

1. [ ] I have read and understood the study description above and understand what will happen during the study.
2. [ ] I understand that I can withdraw at any time before, during or after the study
3. [ ] I agree for data to be taken and used as part of the study
4. [ ] Any video of me CAN be used in presentations without anonymisation

Name:...

Age:..

Height:..

Signature:...

Date:...

1. For each selection technique you used please indicate how much you like the technique (1 strongly dislike and 10 strongly like)

Selection Technique	1	2	3	4	5	6	7	8	9	10
1.Reach with feedback										
2.Reach without feedback										
3.Direction with feedback										
4.Direction without feedback										

Please give reasons

2. For each direction you selected please indicate how comfortable you feel when selecting in the direction (1 very uncomfortable and 10 very comfortable)

Selection Direction		1	2	3	4	5	6	7	8	9	10
1.Up	↑										
2.Up-right	↗										
3.Right	→										
4.Down-right	↘										
5.Down	↓										
6.Down-left	↙										
7.Left	←										
8.Up-left	↖										

Please give reasons

3. For the errors you made in the experiment, please indicate the reason of the errors

Selection Technique	Reach with feedback	Reach without feedback	Direction with feedback	Direction without feedback
Reasons				

Please give more reasons of your errors here:

____

____

____

____

____

____

Participant No...............

Gesture Selection Study

Thank you for taking part in this study! The aim of the study is to gain an insight into the usability and performance aspects of gestural interaction in 3D selection task. The study will be conducted for near one hour. And contain four sessions and one questionnaire in the end. Each session has a training block and a test block.You can withdraw at any time before, during or after the study.

Data Collection For This Study: During the study we will be filming you. This video may be used in presentations. Anonymisation of video is not easy, so, if you are willing for your video to be shown in presentations then please indicate below.

Additional Data Collection: In addition to being filmed we will be recording data from a Kinect camera. We are collecting this data for study the patterns of body movement during gestures.

Handling of Data: All data will as far as possible be anonymised – video data can only be identified by your image. All data will be stored securely.

1. [ ]I have read and understood the study description above and understand what will happen during the study.
2. [ ]I understand that I can withdraw at any time before, during or after the study
3. [ ]I agree for data to be taken and used as part of the study
4. [ ]Any video of me CAN be used in presentations without anonymisation

Name:..

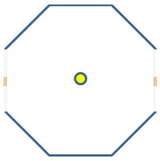
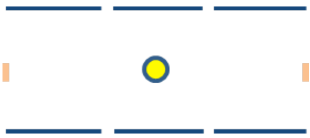
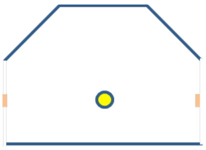
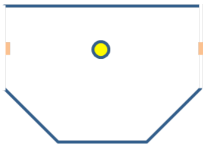
Age:...

Height:..

Signature:...

Date:...

1. For each selection technique you used please indicate how much you like the technique (1 strongly dislike and 10 strongly like)

3D menu shape	1	2	3	4	5	6	7	8	9	10
 0.										
 1.										
 2.										
 3.										

Please give reasons

2. For each direction you selected please indicate how comfortable you feel when selecting in the direction (1 very uncomfortable and 10 very comfortable)

Selection Direction (X-Y Plane)		1	2	3	4	5	6	7	8	9	10
1.Up	↑										
2.Up-right	↗										
3.Right	→										
4.Down-right	↘										
5.Down	↓										
6.Down-left	↙										
7.Left	←										
8.Up-left	↖										

Selection Direction (Z axis)		1	2	3	4	5	6	7	8	9	10
9. Back-right	↘										
10. Back	↓										
11. Back-left	↙										
12. Front-left	↖										
13. Front	↑										
14. Front-right	↗										

Please give reasons

Participant No...............

Gesture Input Study

Thank you for taking part in this study! The aim of the study is to gain an insight into the usability and performance aspects of gestural text input task. The study will be conducted for near half hour each day and last for a week. Each session contains six blocks. You can withdraw at any time before, during or after the study.

Data Collection For This Study: During the study we will be filming you. This video may be used in presentations. Anonymisation of video is not easy, so, if you are willing for your video to be shown in presentations then please indicate below.

Additional Data Collection: In addition to being filmed we will be recording data from a Kinect camera. We are collecting this data for study the patterns of body movement during gestures.

Handling of Data: All data will as far as possible be anonymised – video data can only be identified by your image. All data will be stored securely.

1. [ ]I have read and understood the study description above and understand what will happen during the study.
2. [ ]I understand that I can withdraw at any time before, during or after the study
3. [ ]I agree for data to be taken and used as part of the study

Name:...

Age:..

Signature:..

Date:...

1. QUERTY Timeout

Mental Demand — How mentally demanding was the task?

Very Low — Very High

Physical Demand — How physically demanding was the task?

Very Low — Very High

Temporal Demand — How hurried or rushed was the pace of the task?

Very Low — Very High

Performance — How successful were you in accomplishing what you were asked to do?

Perfect — Failure

Effort — How hard did you have to work to accomplish your level of performance?

Very Low — Very High

Frustration — How insecure, discouraged, irritated, stressed, and annoyed wereyou?

Very Low — Very High

2. QUERTY Reach

Mental Demand — How mentally demanding was the task?

Very Low — Very High

Physical Demand — How physically demanding was the task?

Very Low — Very High

Temporal Demand — How hurried or rushed was the pace of the task?

Very Low — Very High

Performance — How successful were you in accomplishing what you were asked to do?

Perfect — Failure

Effort — How hard did you have to work to accomplish your level of performance?

Very Low — Very High

Frustration — How insecure, discouraged, irritated, stressed, and annoyed wereyou?

Very Low — Very High

3. QUERTY Expand and Reach

Mental Demand

How mentally demanding was the task?

Very Low — Very High

Physical Demand

How physically demanding was the task?

Very Low — Very High

Temporal Demand

How hurried or rushed was the pace of the task?

Very Low — Very High

Performance

How successful were you in accomplishing what you were asked to do?

Perfect — Failure

Effort

How hard did you have to work to accomplish your level of performance?

Very Low — Very High

Frustration

How insecure, discouraged, irritated, stressed, and annoyed wereyou?

Very Low — Very High

4. Circle Timeout

Mental Demand

How mentally demanding was the task?

Very Low | Very High

Physical Demand

How physically demanding was the task?

Very Low | Very High

Temporal Demand

How hurried or rushed was the pace of the task?

Very Low | Very High

Performance

How successful were you in accomplishing what you were asked to do?

Perfect | Failure

Effort

How hard did you have to work to accomplish your level of performance?

Very Low | Very High

Frustration

How insecure, discouraged, irritated, stressed, and annoyed wereyou?

Very Low | Very High

5. Circle Reach

Mental Demand — How mentally demanding was the task?

Very Low — Very High

Physical Demand — How physically demanding was the task?

Very Low — Very High

Temporal Demand — How hurried or rushed was the pace of the task?

Very Low — Very High

Performance — How successful were you in accomplishing what you were asked to do?

Perfect — Failure

Effort — How hard did you have to work to accomplish your level of performance?

Very Low — Very High

Frustration — How insecure, discouraged, irritated, stressed, and annoyed wereyou?

Very Low — Very High

6. Circle Expand and Reach

Mental Demand

How mentally demanding was the task?

Very Low | Very High

Physical Demand

How physically demanding was the task?

Very Low | Very High

Temporal Demand

How hurried or rushed was the pace of the task?

Very Low | Very High

Performance

How successful were you in accomplishing what you were asked to do?

Perfect | Failure

Effort

How hard did you have to work to accomplish your level of performance?

Very Low | Very High

Frustration

How insecure, discouraged, irritated, stressed, and annoyed wereyou?

Very Low | Very High

Questionnaire **Participant No :.........................**

1. For each selection technique you used please indicate how much you like the technique (1 strongly dislike and 10 strongly like)

Selection Technique	1	2	3	4	5	6	7	8	9	10
1.QUERTY Timeout										
2.QUERTY Reach										
3.QUERTY Expand and Reach										
4.Circle Timeout										
5.Circle Reach										
6.Circle Expand and Reach										

Please give reasons

BIBLIOGRAPHY

[1] 5dt data glove manual. Technical report, Fifth Dimension Technologies.

[2] J. Accot and S. Zhai. More than dotting the i's — foundations for crossing-based interfaces. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, CHI '02, pages 73–80, New York, NY, USA, 2002. ACM.

[3] J. Alexander, M. T. Marshall, and S. Subramanian. Adding haptic feedback to mobile tv. In *CHI '11 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '11, pages 1975–1980, New York, NY, USA, 2011. ACM.

[4] R. Aoki, B. Chan, M. Ihara, T. Kobayashi, M. Kobayashi, and S. Kagami. A gesture recognition algorithm for vision-based unicursal gesture interfaces. In *Proceedings of the 10th European Conference on Interactive TV and Video*, EuroiTV '12, pages 53–56, New York, NY, USA, 2012. ACM.

[5] C. Appert and S. Zhai. Using strokes as command shortcuts: cognitive benefits and toolkit support. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '09, pages 2289–2298, New York, NY, USA, 2009. ACM.

[6] D. Ashbrook and T. Starner. Magic: a motion gesture design tool. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, pages 2159–2168, New York, NY, USA, 2010. ACM.

[7] S.-H. Bae, R. Balakrishnan, and K. Singh. Ilovesketch: as-natural-as-possible sketching system for creating 3d curve models. In *Proceedings of the 21st annual ACM symposium on User interface software and technology*, UIST '08, pages 151–160, New York, NY, USA, 2008. ACM.

[8] N. Bakker, P. Werkhoven, and P. Passenier. Aiding orientation performance in virtual environments with proprioceptive feedback. In *Virtual Reality Annual*

International Symposium, 1998. Proceedings., IEEE 1998, pages 28 –33, 18-18 1998.

[9] M. Baldauf, S. Zambanini, P. Fröhlich, and P. Reichl. Markerless visual fingertip detection for natural mobile device interaction. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services*, MobileHCI '11, pages 539–544, New York, NY, USA, 2011. ACM.

[10] R. Ball, C. North, and D. A. Bowman. Move to improve: promoting physical navigation to increase user performance with large displays. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, CHI '07, pages 191–200, New York, NY, USA, 2007. ACM.

[11] F. A. Barrientos and J. F. Canny. Cursive:: controlling expressive avatar gesture using pen gesture. In *Proceedings of the 4th international conference on Collaborative virtual environments*, CVE '02, pages 113–119, New York, NY, USA, 2002. ACM.

[12] M. Beaudouin-Lafon. Lessons learned from the wild room, a multisurface interactive environment. In *23rd French Speaking Conference on Human-Computer Interaction*, IHM '11, pages 18:1–18:8, New York, NY, USA, 2011. ACM.

[13] H. Benko. Beyond flat surface computing: challenges of depth-aware and curved interfaces. In *Proceedings of the 17th ACM international conference on Multimedia*, MM '09, pages 935–944, New York, NY, USA, 2009. ACM.

[14] H. Benko, R. Jota, and A. Wilson. Miragetable: freehand interaction on a projected augmented reality tabletop. In *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems*, CHI '12, pages 199–208, New York, NY, USA, 2012. ACM.

[15] H. Benko and A. Wilson. Depthtouch: Using depth-sensing camera to enable freehand interactions on and above the interactive surface. Technical report, Microsoft Research Technical Report MSR-TR-2009-23, 2009.

[16] H. Benko and A. D. Wilson. Multi-point interactions with immersive omnidirectional visualizations in a dome. In *ACM International Conference on Interactive Tabletops and Surfaces*, ITS '10, pages 19–28, New York, NY, USA, 2010. ACM.

[17] H. Benko, A. D. Wilson, and P. Baudisch. Precise selection techniques for multi-touch screens. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, CHI '06, pages 1263–1272, New York, NY, USA, 2006. ACM.

[18] X. Bi, B. A. Smith, and S. Zhai. Quasi-qwerty soft keyboard optimization. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, pages 283–286, New York, NY, USA, 2010. ACM.

[19] B. Bideau, R. Kulpa, N. Vignais, S. Brault, F. Multon, and C. Craig. Using virtual reality to analyze sports performance. *Computer Graphics and Applications, IEEE*, 30(2):14–21, 2010.

[20] J. Blake. *Natural user interfaces in .net.* Manning Publications Co, 2011.

[21] J. Bobeth, S. Schmehl, E. Kruijff, S. Deutsch, and M. Tscheligi. Evaluating performance and acceptance of older adults using freehand gestures for tv menu control. In *Proceedings of the 10th European Conference on Interactive TV and Video*, EuroiTV '12, pages 35–44, New York, NY, USA, 2012. ACM.

[22] T. Bockemuhl, N. F. Troje, and V. Durr. Inter-joint coupling and joint angle synergies of human catching movements. *Human Movement Science*, 29(1):73 – 93, 2010.

[23] R. A. Bolt. Put-that-there : Voice and gesture at the graphics interface. In *Proceedings of the 7th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '80, pages 262–270, New York, NY, USA, 1980. ACM.

[24] D. Bowman and C. Wingrave. Design and evaluation of menu systems for immersive virtual environments. In *IEEE Virtual Reality*, pages 149 –156, march 2001.

[25] D. A. Bowman, J. Chen, C. A. Wingrave, J. F. Lucas, A. Ray, N. F. Polys, Q. Li, Y. Haciahmetoglu, J.-S. Kim, S. Kim, et al. New directions in 3d user interfaces. *IJVR*, 5(2):3–14, 2006.

[26] D. A. Bowman, S. Coquillart, B. Froehlich, M. Hirose, Y. Kitamura, K. Kiyokawa, and W. Stuerzlinger. 3d user interfaces: New directions and perspectives. *Computer Graphics and Applications, IEEE*, 28(6):20–36, 2008.

[27] D. A. Bowman and L. F. Hodges. An evaluation of techniques for grabbing and manipulating remote objects in immersive virtual environments. In *Proceedings of the 1997 symposium on Interactive 3D graphics*, I3D '97, pages 35–ff., New York, NY, USA, 1997. ACM.

[28] D. A. Bowman and L. F. Hodges. Formalizing the design, evaluation, and application of interaction techniques for immersive virtual environments. *Journal of Visual Languages and Computing*, 10(1):37 – 53, 1999.

[29] A. Bragdon and H.-S. Ko. Gesture select: acquiring remote targets on large displays without pointing. In *Proceedings of the 2011 annual conference on Human factors in computing systems*, CHI '11, pages 187–196, New York, NY, USA, 2011. ACM.

[30] W. Broll, J. Herling, and L. Blum. Interactive bits: Prototyping of mixed reality applications and interaction techniques through visual programming. In *3D User Interfaces, 2008. 3DUI 2008. IEEE Symposium on*, pages 109 –115, march 2008.

[31] V. Buchmann, S. Violich, M. Billinghurst, and A. Cockburn. Fingartips: gesture based direct manipulation in augmented reality. In *Proceedings of the 2nd international conference on Computer graphics and interactive techniques in Australasia and South East Asia*, pages 212–221. ACM, 2004.

[32] M. C. Cabral, C. H. Morimoto, and M. K. Zuffo. On the usability of gesture interfaces in virtual reality environments. In *Proceedings of the 2005 Latin American conference on Human-computer interaction*, pages 100–108. ACM, 2005.

[33] J. Callahan, D. Hopkins, M. Weiser, and B. Shneiderman. An empirical comparison of pie vs. linear menus. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, CHI '88, pages 95–100, New York, NY, USA, 1988. ACM.

[34] X. Cao and R. Balakrishnan. Visionwand: interaction techniques for large displays using a passive wand tracked in 3d. In *Proceedings of the 16th annual ACM symposium on User interface software and technology*, UIST '03, pages 173–182, New York, NY, USA, 2003. ACM.

[35] S. K. Card, J. D. Mackinlay, and G. G. Robertson. A morphological analysis of the design space of input devices. *ACM Transactions on Information Systems (TOIS)*, 9(2):99–122, Apr. 1991.

[36] M. Carpendale, D. Cowperthwaite, and F. Fracchia. Extending distortion viewing from 2d to 3d. *Computer Graphics and Applications, IEEE*, 17(4):42 –51, jul/aug 1997.

[37] S. J. Castellucci and I. S. MacKenzie. Gathering text entry metrics on android devices. In *CHI '11 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '11, pages 1507–1512, New York, NY, USA, 2011. ACM.

[38] J. Cechanowicz, S. Dawson, M. Victor, and S. Subramanian. Stylus based text input using expanding cirrin. In *Proceedings of the working conference on Advanced visual interfaces*, AVI '06, pages 163–166, New York, NY, USA, 2006. ACM.

[39] D. Chatterjee, A. Sinha, A. Pal, and A. Basu. An iterative methodolgy to improve tv onscreen keyboard layout design through evaluation of user studies. *Advances in Computing*, 2(5):81–91, 2012.

[40] S. Chatty and P. Lecoanet. Pen computing for air traffic control. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '96, pages 87–94, New York, NY, USA, 1996. ACM.

[41] D. B. Chertoff, R. W. Byers, and J. J. LaViola, Jr. An exploration of menu techniques using a 3d game input device. In *Proceedings of the 4th International Conference on Foundations of Digital Games*, FDG '09, pages 256–262, New York, NY, USA, 2009. ACM.

[42] G. Cirio, P. Vangorp, E. Chapoulie, M. Marchal, A. Lecuyer, and G. Drettakis. Walking in a cube: Novel metaphors for safely navigating large virtual environments in restricted real workspaces. *Visualization and Computer Graphics, IEEE Transactions on*, 18(4):546–554, 2012.

[43] A. Clark, A. Dünser, M. Billinghurst, T. Piumsomboon, and D. Altimira. Seamless interaction in space. In *Proceedings of the 23rd Australian Computer-Human Interaction Conference*, OzCHI '11, pages 88–97, New York, NY, USA, 2011. ACM.

[44] P. Cohen, D. McGee, S. Oviatt, L. Wu, J. Clow, R. King, S. Julier, and L. Rosenblum. Multimodal interaction for 2d and 3d environments [virtual reality]. *Computer Graphics and Applications, IEEE*, 19(4):10–13, 1999.

[45] P. R. Cohen, M. Johnston, D. McGee, S. Oviatt, J. Pittman, I. Smith, L. Chen, and J. Clow. Quickset: multimodal interaction for distributed applications. In *Proceedings of the fifth ACM international conference on Multimedia*, MULTIMEDIA '97, pages 31–40, New York, NY, USA, 1997. ACM.

[46] C. Cruz-Neira, D. J. Sandin, and T. A. DeFanti. Surround-screen projection-based virtual reality: the design and implementation of the cave. In *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '93, pages 135–142, New York, NY, USA, 1993. ACM.

[47] R. Dachselt and A. Hübner. Three-dimensional menus: A survey and taxonomy. *Computers & Graphics*, 31(1):53 – 65, 2007.

[48] K. Das and C. Borst. An evaluation of menu properties and pointing techniques in a projection-based vr environment. In *2010 IEEE Symposium on 3D User Interfaces*, pages 47 –50, march 2010.

[49] L. Deligiannidis and J. Larkin. Navigating inexpensively and wirelessly. In *Human System Interactions, 2008 Conference on*, pages 165 –169, may 2008.

[50] T. Döring, A. S. Shirazi, and A. Schmidt. Exploring gesture-based interaction techniques in multi-display environments with mobile phones and a multi-touch table. In *Proceedings of the International Conference on Advanced Visual Interfaces*, AVI '10, pages 419–419, New York, NY, USA, 2010. ACM.

[51] M. Doulis, V. Zwimpfer, J. Pfluger, A. Simon, C. Stern, T. Haldimann, and C. Jenni. Spaceactor - interface prototypes for virtual environments. In *3D User Interfaces, 2006. 3DUI 2006. IEEE Symposium on*, pages 171 – 174, march 2006.

[52] M. D. Dunlop, N. Durga, S. Motaparti, P. Dona, and V. Medapuram. Qwerth: an optimized semi-ambiguous keyboard design. In *Proceedings of the 14th International Conference on Human-Computer Interaction with Mobile devices and Services*, MobileHCI '12, pages 23–28, New York, NY, USA, 2012. ACM.

[53] T. Duval, A. Lecuyer, and S. Thomas. Skewer: a 3d interaction technique for 2-user collaborative manipulation of objects in virtual environments. In *3D User Interfaces, 2006. 3DUI 2006. IEEE Symposium on*, pages 69–72, 2006.

[54] F. Farhadi-Niaki, S. Etemad, and A. Arya. Design and usability analysis of gesture-based control for common desktop tasks. In M. Kurosu, editor, *Human-Computer Interaction. Interaction Modalities and Techniques*, volume 8007 of *Lecture Notes in Computer Science*, pages 215–224. Springer Berlin Heidelberg, 2013.

[55] L. Findlater, A. Jansen, K. Shinohara, M. Dixon, P. Kamb, J. Rakita, and J. O. Wobbrock. Enhanced area cursors: reducing fine pointing demands for people with motor impairments. In *Proceedings of the 23rd Annual ACM Symposium on User Interface Software and Technology*, UIST '10, pages 153–162, New York, NY, USA, 2010. ACM.

[56] S. S. Fisher, M. McGreevy, J. Humphries, and W. Robinett. Virtual environment display system. In *Proceedings of the 1986 workshop on Interactive 3D graphics*, I3D '86, pages 77–87, New York, NY, USA, 1987. ACM.

[57] P. M. Fitts. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of experimental psychology*, 47(6):381, 1954.

[58] G. Fitzmaurice, J. Matejka, A. Khan, M. Glueck, and G. Kurtenbach. Piecursor: merging pointing and command selection for rapid in-place tool switching. In *Proceedings of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, CHI '08, pages 1361–1370, New York, NY, USA, 2008. ACM.

[59] G. W. Fitzmaurice, J. Matejka, I. Mordatch, A. Khan, and G. Kurtenbach. Safe 3d navigation. In *SI3D*, pages 7–15, 2008.

[60] T. Flash and N. Hogan. The coordination of arm movements: an experimentally confirmed mathematical model. *The journal of Neuroscience*, 5(7):1688–1703, 1985.

[61] S. Frees and G. D. Kessler. Precise and rapid interaction through scaled manipulation in immersive virtual environments. In *Proceedings of the 2005 IEEE Conference 2005 on Virtual Reality*, VR '05, pages 99–106, Washington, DC, USA, 2005. IEEE Computer Society.

[62] L. Gallo and M. Ciampi. Wii remote-enhanced hand-computer interaction for 3d medical image analysis. In *Current Trends in Information Technology (CTIT), 2009 International Conference on the*, pages 1–6, 2009.

[63] M. Gately, Y. Zhai, M. Yeary, E. Petrich, and L. Sawalha. A three-dimensional swept volume display based on led arrays. *J. Display Technol.*, 7(9):503–514, Sep 2011.

[64] G. Geleijnse, D. Aliakseyeu, and E. Sarroukh. Comparing text entry methods for interactive television applications. In *Proceedings of the 7th European Conference on Interactive TV and Video*, EuroiTV '09, pages 145–148, New York, NY, USA, 2009. ACM.

[65] A. P. Georgopoulos, J. F. Kalaska, R. Caminiti, and J. T. Massey. On the relations between the direction of two-dimensional arm movements and cell discharge in primate motor cortex. *The Journal of Neuroscience*, 2(11):1527–1537, 1982.

[66] M. Glueck, S. Anderson, and A. Khan. Deskcube: using physical zones to select and control combinations of 3d navigation operations. In *SpringSim*, page 200, 2010.

[67] J. Grosjean, J.-M. Burkhardt, S. Coquillart, and P. Richard. Evaluation of the command and control cube. In *Multimodal Interfaces, 2002. Proceedings. Fourth IEEE International Conference on*, pages 473 – 478, 2002.

[68] T. Grossman and R. Balakrishnan. Pointing at trivariate targets in 3d environments. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '04, pages 447–454, New York, NY, USA, 2004. ACM.

[69] T. Grossman and R. Balakrishnan. The design and evaluation of selection techniques for 3d volumetric displays. In *Proceedings of the 19th annual ACM symposium on User interface software and technology*, UIST '06, pages 3–12, New York, NY, USA, 2006. ACM.

[70] T. Grossman, D. Wigdor, and R. Balakrishnan. Multi-finger gestural interaction with 3d volumetric displays. In *Proceedings of the 17th annual ACM symposium on User interface software and technology*, UIST '04, pages 61–70, New York, NY, USA, 2004. ACM.

[71] F. Guimbretière and C. Nguyen. Bimanual marking menu for near surface interactions. In *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems*, CHI '12, pages 825–828, New York, NY, USA, 2012. ACM.

[72] F. Guimbretiére and T. Winograd. Flowmenu: combining command, text, and data entry. In *Proceedings of the 13rd Annual ACM Symposium on User Interface Software and Technology*, UIST '00, pages 213–216. ACM, 2000.

[73] F. Guimbretiére and T. Winograd. Flowmenu: combining command, text, and data entry. In *Proceedings of the 13th annual ACM symposium on User interface software and technology*, UIST '00, pages 213–216, New York, NY, USA, 2000. ACM.

[74] C. Harrison, H. Benko, and A. D. Wilson. Omnitouch: wearable multitouch interaction everywhere. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, UIST '11, pages 441–450, New York, NY, USA, 2011. ACM.

[75] C. Harrison, D. Tan, and D. Morris. Skinput: appropriating the body as an input surface. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, pages 453–462, New York, NY, USA, 2010. ACM.

[76] S. G. Hart. Nasa-task load index (nasa-tlx); 20 years later. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 50, pages 904–908. Sage Publications, 2006.

[77] S. G. Hart and L. E. Staveland. Development of nasa-tlx (task load index): Results of empirical and theoretical research. *Advances in psychology*, 52:139–183, 1988.

[78] N. Henze, E. Rukzio, and S. Boll. Observational and experimental investigation of typing behaviour using virtual keyboards for mobile devices. In *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems*, CHI '12, pages 2659–2668, New York, NY, USA, 2012. ACM.

[79] L. Herda, P. Fua, R. Plänkers, R. Boulic, and D. Thalmann. Using skeleton-based tracking to increase the reliability of optical motion capture. *Human Movement Science*, 20(3):313 – 341, 2001.

[80] U. Hinrichs and S. Carpendale. Gestures in the wild: studying multi-touch gesture sequences on interactive tabletop exhibits. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, CHI '11, pages 3023–3032, New York, NY, USA, 2011. ACM.

[81] K. Hornbaek and M. Hertzum. Untangling the usability of fisheye menus. *ACM Trans. Comput.-Hum. Interact.*, 14(2), Aug. 2007.

[82] T. Hoshi, T. Iwamoto, and H. Shinoda. Non-contact tactile sensation synthesized by ultrasound transducers. In *EuroHaptics conference, 2009 and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems. World Haptics 2009. Third Joint*, pages 256–260, 2009.

[83] L. Hoste, B. Dumas, and B. Signer. Speeg: a multimodal speech- and gesture-based text input solution. In *Proceedings of the International Working Conference on Advanced Visual Interfaces*, AVI '12, pages 156–163, New York, NY, USA, 2012. ACM.

[84] W. Hurst and C. Wezel. Gesture-based interaction via finger tracking for mobile augmented reality. *Multimedia Tools and Applications*, 62(1):233–258, 2013.

[85] A. Iatrino and S. Modeo. Text editing in digital terrestrial television: a comparison of three interfaces. In *Proceedings of the 4th European Conference on Interactive TV and Video*, EuroiTV '06. Citeseer, 2006.

[86] V. Interrante, B. Ries, and L. Anderson. Seven league boots: A new metaphor for augmented locomotion through moderately large scale immersive virtual environments. In *3D User Interfaces, 2007. 3DUI '07. IEEE Symposium on*, pages –, 2007.

[87] S. Irawati, S. Green, M. Billinghurst, A. Duenser, and H. Ko. An evaluation of an augmented reality multimodal interface using speech and paddle gestures. In *Advances in Artificial Reality and Tele-Existence*, pages 272–283. Springer, 2006.

[88] P. Isokoski and R. Raisamo. Quikwriting as a multi-device text entry method. In *Proceedings of the Third Nordic Conference on Human-Computer Interaction*, NordiCHI '04, pages 105–108, New York, NY, USA, 2004. ACM.

[89] C. Jetter, J. Gerken, and H. Reiterer. Natural user interfaces: Why we need better model-worlds, not better gestures. In *CHI 2010 Workshop - Natural User Interfaces: The Prospect and Challenge of Touch and Gestural Computing*, pages 1–4, 2010.

[90] Y. Jin, S. Choi, A. Chung, I. Myung, J. Lee, M. Kim, and J. Woo. Gia: design of a gesture-based interaction photo album. *Personal and Ubiquitous Computing*, 8(3-4):227–233, 2004.

[91] H. Jo, S. Hwang, H. Park, and J. hee Ryu. Aroundplot: Focus+context interface for off-screen objects in 3d environments. *Computers & Graphics*, 35(4):841 – 853, 2011.

[92] E. Jones, J. Alexander, A. Andreou, P. Irani, and S. Subramanian. Gestext: accelerometer-based gestural text-entry systems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, pages 2173–2182. ACM, 2010.

[93] M. Karam. *PhD Thesis: A framework for research and design of gesture-based human-computer interactions.* PhD thesis, University of Southampton, 2006.

[94] M. Karam and m. c. schraefel. A taxonomy of gestures in human computer interactions. Technical report, University of Southampton, 2005.

[95] S. Kettebekov and R. Sharma. Toward natural gesture/speech control of a large display. In M. Little and L. Nigay, editors, *Engineering for Human-Computer Interaction*, volume 2254 of *Lecture Notes in Computer Science*, pages 221–234. Springer Berlin Heidelberg, 2001.

[96] A. Khan, I. Mordatch, G. Fitzmaurice, J. Matejka, and G. Kurtenbach. Viewcube: a 3d orientation indicator and controller. In *Proceedings of the 2008 symposium on Interactive 3D graphics and games*, I3D '08, pages 17–25, New York, NY, USA, 2008. ACM.

[97] M. Kolsch, A. C. Beall, and M. Turk. The postural comfort zone for reaching gestures. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 47, pages 725–728, 2003.

[98] R. Kopper, F. Bacim, and D. Bowman. Rapid and accurate 3d selection by progressive refinement. In *2011 IEEE Symposium on 3D User Interfaces*, pages 67 –74, march 2011.

[99] R. Kopper, T. Ni, D. Bowman, and M. Pinho. Design and evaluation of navigation techniques for multiscale virtual environments. In *Virtual Reality Conference, 2006*, pages 175 – 182, march 2006.

[100] P. O. Kristensson, T. Nicholson, and A. Quigley. Continuous recognition of one-handed and two-handed gestures using 3d full-body motion tracking sensors. In *Proceedings of the 2012 ACM international conference on Intelligent User Interfaces*, IUI '12, pages 89–92, New York, NY, USA, 2012. ACM.

[101] P.-O. Kristensson and S. Zhai. Shark2: a large vocabulary shorthand writing system for pen-based computers. In *Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology*, UIST '04, pages 43–52. ACM, 2004.

[102] K. Kuikkaniemi, G. Jacucci, M. Turpeinen, E. Hoggan, and J. Müller. From space to stage: How interactive screens will change urban life. *Computer*, 44(6):40–47, 2011.

[103] G. Kurtenbach. *The Design and Evaluation of Marking Menus.* PhD thesis, University of Toronto, 1993.

[104] G. Kurtenbach and W. Buxton. Issues in combining marking and direct manipulation techniques. In *Proceedings of the 4th annual ACM symposium on User interface software and technology*, UIST '91, pages 137–144, New York, NY, USA, 1991. ACM.

[105] G. Kurtenbach and W. Buxton. The limits of expert performance using hierarchic marking menus. In *Proceedings of CHI '93 conference on Human factors in computing systems*, CHI '93, pages 482–487, New York, NY, USA, 1993. ACM.

[106] G. Kurtenbach and W. Buxton. User learning and performance with marking menus. In *Proceedings of the SIGCHI conference on Human factors in computing systems: celebrating interdependence*, CHI '94, pages 258–264, New York, NY, USA, 1994. ACM.

[107] G. Kurtenbach, A. J. Sellen, and W. Buxton. An empirical evaluation of some articulatory and cognitive aspects of marking menus. *Human-Computer Interaction*, 8(1):1–23, Mar. 1993.

[108] G. D. Langolf, D. B. Chaffin, and J. A. Foulke. An investigation of fitts' law using a wide range of movement amplitudes. *Journal of Motor Behavior*, 8(2):113–128, 1976.

[109] J. LaViola. Bringing vr and spatial 3d interaction to the masses through video games. *Computer Graphics and Applications, IEEE*, 28(5):10–15, 2008.

[110] B. Lee, P. Isenberg, N. Riche, and S. Carpendale. Beyond mouse and keyboard: Expanding design considerations for information visualization interactions. *Visualization and Computer Graphics, IEEE Transactions on*, 18(12):2689–2698, 2012.

[111] J. Lee. Hacking the nintendo wii remote. *Pervasive Computing, IEEE*, 7(3):39–45, 2008.

[112] W.-P. Lee, C. Kaoli, and J.-Y. Huang. A smart tv system with body-gesture control, tag-based rating and context-aware recommendation. *Knowledge-Based Systems*, 56:167–178, 2014.

[113] G. J. Lepinski, T. Grossman, and G. Fitzmaurice. The design and evaluation of multitouch marking menus. In *Proceedings of the 28th international conference on Human factors in computing systems*, CHI '10, pages 2233–2242, New York, NY, USA, 2010. ACM.

[114] F. C. Y. Li, R. T. Guy, K. Yatani, and K. N. Truong. The 1line keyboard: a qwerty layout in a single line. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, UIST '11, pages 461–470, New York, NY, USA, 2011. ACM.

[115] J. Liang and M. Green. JDCAD: A highly interactive 3D modeling system. In *Computer and Graphics*, volume 18(4), pages 499–506, 1994.

[116] I. S. MacKenzie and R. W. Soukoreff. Phrase sets for evaluating text entry techniques. In *CHI '03 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '03, pages 754–755, New York, NY, USA, 2003. ACM.

[117] P. Majaranta, U.-K. Ahola, and O. Špakov. Fast gaze typing with an adjustable dwell time. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '09, pages 357–360, New York, NY, USA, 2009. ACM.

[118] J. Mankoff and G. D. Abowd. Cirrin: A word-level unistroke keyboard for pen input. In *Proceedings of the 11th Annual ACM Symposium on User Interface Software and Technology*, UIST'98, pages 213–214. ACM, 1998.

[119] T. Margolis, T. A. DeFanti, G. Dawe, A. Prudhomme, J. P. Schulze, and S. Cutchin. Low cost heads-up virtual reality (huvr) with optical tracking and haptic feedback. pages 786417–786417–11, 2011.

[120] J. McCrae, M. Glueck, T. Grossman, A. Khan, and K. Singh. Exploring the design space of multiscale 3d orientation. In *Proceedings of the International Conference on Advanced Visual Interfaces*, AVI '10, pages 81–88, New York, NY, USA, 2010. ACM.

[121] J. McCrae, I. Mordatch, M. Glueck, and A. Khan. Multiscale 3d navigation. In *SI3D*, pages 7–14, 2009.

[122] M. J. McGuffin and R. Balakrishnan. Fitts' law and expanding targets: Experimental studies and designs for user interfaces. *ACM TOCHI*, 12(4):388–422, 2005.

[123] H. M. Mentis, K. O'Hara, A. Sellen, and R. Trivedi. Interaction proxemics and image use in neurosurgery. In *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems*, CHI '12, pages 927–936, New York, NY, USA, 2012. ACM.

[124] A. Murata and H. Iwase. Extending fitts' law to a three-dimensional pointing task. *Human Movement Science*, 20(6):791 – 805, 2001.

[125] E. Murphy-Chutorian and M. Trivedi. Head pose estimation and augmented reality tracking: An integrated system and evaluation for monitoring driver awareness. *Intelligent Transportation Systems, IEEE Transactions on*, 11(2):300–311, 2010.

[126] T. Ni, D. Bowman, and C. North. Airstroke: bringing unistroke text entry to freehand gesture interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 2473–2476. ACM, 2011.

[127] T. Ni, D. A. Bowman, C. North, and R. P. McMahan. Design and evaluation of freehand menu selection interfaces using tilt and pinch gestures. *International Journal of Human-Computer Studies*, 69:551–562, August 2011.

[128] T. Ni, D. A. Bowman, C. North, and R. P. McMahan. Design and evaluation of freehand menu selection interfaces using tilt and pinch gestures. *International Journal of Human-Computer Studies*, 69:551–562, August 2011.

[129] D. A. Norman. Natural user interfaces are not natural. *Interactions*, 17(3):6–10, May 2010.

[130] J. Ou, S. R. Fussell, X. Chen, L. D. Setlock, and J. Yang. Gestural communication over video stream: supporting multimodal interaction for remote collaborative physical tasks. In *Proceedings of the 5th international conference on Multimodal interfaces*, ICMI '03, pages 242–249, New York, NY, USA, 2003. ACM.

[131] S. Oviatt. Ten myths of multimodal interaction. *Commun. ACM*, 42(11):74–81, Nov. 1999.

[132] S. Oviatt and P. Cohen. Perceptual user interfaces: multimodal interfaces that process what comes naturally. *Commun. ACM*, 43(3):45–53, Mar. 2000.

[133] K. Perlin. Quikwriting: continuous stylus-based text entry. In *Proceedings of the 11th Annual ACM Symposium on User Interface Software and Technology*, UIST '98, pages 215–216, New York, NY, USA, 1998. ACM.

[134] O. Polacek, M. Klima, A. J. Sporka, P. Zak, M. Hradis, P. Zemcik, and V. Prochazka. A comparative study on distant free-hand pointing. In *Proceedings of*

the 10th European Conference on Interactive TV and Video, EuroiTV '12, pages 139–142, New York, NY, USA, 2012. ACM.

[135] Primesense. The primesense 3d awareness sensor.

[136] M. Rahman, S. Gustafson, P. Irani, and S. Subramanian. Tilt techniques: investigating the dexterity of wrist-based input. In *Proceedings of the 27th international conference on Human factors in computing systems*, CHI '09, pages 1943–1952, New York, NY, USA, 2009. ACM.

[137] R. Raskar, G. Welch, M. Cutts, A. Lake, L. Stesin, and H. Fuchs. The office of the future: a unified approach to image-based modeling and spatially immersive displays. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '98, pages 179–188, New York, NY, USA, 1998. ACM.

[138] V. Reddy, V. Raghuveer, J. Krishna, and K. Chandralohit. Finger gesture based tablet interface. In *Computational Intelligence Computing Research (ICCIC), 2012 IEEE International Conference on*, pages 1–4, 2012.

[139] J. Rekimoto. Smartskin: an infrastructure for freehand manipulation on interactive surfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '02, pages 113–120, New York, NY, USA, 2002. ACM.

[140] F. Remondino and A. Roditakis. Human motion reconstruction and animation from video sequences. In *17th International Conference on Computer Animation and Social Agents (CASA2004)*, pages 347–354. Citeseer, 2004.

[141] G. Ren and E. O'Neill. 3d marking menu selection with freehand gestures. In *IEEE Symposium on 3D User Interfaces*, 3DUI '12, pages 61–68, 2012.

[142] G. Ren and E. O'Neill. 3d selection with freehand gesture. *Computers & Graphics*, 37(3):101 – 120, 2013.

[143] J. Rick. Performance optimizations of virtual keyboards for stroke-based text entry on a touch-based tabletop. In *Proceedings of the 23nd Annual ACM Symposium on User Interface Software and Technology*, UIST '10, pages 77–86, New York, NY, USA, 2010. ACM.

[144] D. Rubine. Combining gestures and direct manipulation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '92, pages 659–660, New York, NY, USA, 1992. ACM.

[145] J. Ruiz and Y. Li. Doubleflip: a motion gesture delimiter for mobile interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 2717–2720, New York, NY, USA, 2011. ACM.

[146] J. Ruiz, Y. Li, and E. Lank. User-defined motion gestures for mobile interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 197–206, New York, NY, USA, 2011. ACM.

[147] Samsung. S pen sdk. online, http://developer.samsung.com/s-pen-sdk.

[148] M. Sato, I. Poupyrev, and C. Harrison. Touché: enhancing touch interaction on humans, screens, liquids, and everyday objects. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pages 483–492, New York, NY, USA, 2012. ACM.

[149] R. Scheibe, M. Moehring, and B. Froehlich. Tactile feedback at the finger tips for improved direct interaction in immersive environments. In *3D User Interfaces, 2007. 3DUI '07. IEEE Symposium on*, march 2007.

[150] G. Schmidt, Y. Baillot, D. Brown, E. Tomlin, and J. Swan. Toward disambiguating multiple selections for frustum-based pointing. In *2006 IEEE Symposium on 3D User Interfaces*, pages 87 – 94, march 2006.

[151] J. Segen and S. Kumar. Gesture vr: vision-based 3d hand interace for spatial interaction. In *Proceedings of the sixth ACM international conference on Multimedia*, MULTIMEDIA '98, pages 455–464, New York, NY, USA, 1998. ACM.

[152] J. Segen and S. Kumar. Video acquired gesture interfaces for the handicapped. In *Proceedings of the sixth ACM international conference on Multimedia: Face/gesture recognition and their applications*, MULTIMEDIA '98, pages 45–48, New York, NY, USA, 1998. ACM.

[153] W. R. Sherman and A. B. Craig. *Understanding virtual reality: Interface, application, and design.* Access Online via Elsevier, 2002.

[154] A. S. Shirazi, T. Döring, P. Parvahan, B. Ahrens, and A. Schmidt. Poker surface: combining a multi-touch table and mobile phones in interactive card games. In *Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services*, MobileHCI '09, pages 73:1–73:2, New York, NY, USA, 2009. ACM.

[155] G. Shoemaker, L. Findlater, J. Q. Dawson, and K. S. Booth. Mid-air text input techniques for very large wall displays. In *Proceedings of Graphics Interface 2009*, GI '09, pages 231–238, Toronto, Ont., Canada, Canada, 2009. Canadian Information Processing Society.

[156] J. Song, W. Kim, H. Son, J. Yoo, J. Kim, R. Kim, and J. Oh. Design and implementation of a remote control for iptv with sensors. In T.-h. Kim, H. Adeli,

D. Slezak, F. Sandnes, X. Song, K.-i. Chung, and K. Arnett, editors, *Future Generation Information Technology*, volume 7105 of *Lecture Notes in Computer Science*, pages 223–228. Springer Berlin Heidelberg, 2011.

[157] P. Song, W. B. Goh, W. Hutama, C.-W. Fu, and X. Liu. A handle bar metaphor for virtual object manipulation with mid-air interaction. In *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems*, CHI '12, pages 1297–1306, New York, NY, USA, 2012. ACM.

[158] P. Song, H. Yu, and S. Winkler. Vision-based 3d finger interactions for mixed reality games with physics simulation. In *Proceedings of The 7th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and Its Applications in Industry*, VRCAI '08, pages 7:1–7:6, New York, NY, USA, 2008. ACM.

[159] R. W. Soukoreff and I. S. MacKenzie. Towards a standard for pointing device evaluation, perspectives on 27 years of fitts' law research in {HCI}. *International Journal of Human-Computer Studies*, 61(6):751 – 789, 2004.

[160] A. J. Sporka, O. Polacek, and P. Slavik. Comparison of two text entry methods on interactive tv. In *Proceedings of the 10th European Conference on Interactive TV and Video*, EuroiTV '12, pages 49–52, New York, NY, USA, 2012. ACM.

[161] S. Stellmach and R. Dachselt. Investigating gaze-supported multimodal pan and zoom. In *Proceedings of the Symposium on Eye Tracking Research and Applications*, ETRA '12, pages 357–360, New York, NY, USA, 2012. ACM.

[162] D. J. Sturman, D. Zeltzer, and S. Pieper. Hands-on interaction with virtual environments. In *Proceedings of the 2nd annual ACM SIGGRAPH symposium on User interface software and technology*, UIST '89, pages 19–24, New York, NY, USA, 1989. ACM.

[163] Y. Takeoka, T. Miyaki, and J. Rekimoto. Z-touch: an infrastructure for 3d gesture interaction in the proximity of tabletop surfaces. In *ACM International Conference on Interactive Tabletops and Surfaces*, ITS '10, pages 91–94, New York, NY, USA, 2010. ACM.

[164] D. S. Tan, G. G. Robertson, and M. Czerwinski. Exploring 3d navigation: combining speed-coupled flying with orbiting. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, CHI '01, pages 418–425, New York, NY, USA, 2001. ACM.

[165] R. Teather and W. Stuerzlinger. Assessing the effects of orientation and device on (constrained) 3d movement techniques. In *IEEE Symposium on 3D User Interfaces*, 3DUI '08, pages 43–50, march 2008.

[166] F. Tian, L. Xu, H. Wang, X. Zhang, Y. Liu, V. Setlur, and G. Dai. Tilt menu: using the 3d orientation information of pen devices to extend the selection capability of pen-based user interfaces. In *Proceedings of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, CHI '08, pages 1371–1380, New York, NY, USA, 2008. ACM.

[167] O. Tuisku, V. Surakka, T. Vanhala, V. Rantanen, and J. Lekkala. Wireless face interface: Using voluntary gaze direction and facial muscle activations for human–computer interaction. *Interacting with Computers*, 24(1):1 – 9, 2012.

[168] L. Vanacken, T. Grossman, and K. Coninx. Exploring the effects of environment density and target visibility on object selection in 3d virtual environments. In *2007 IEEE Symposium on 3D User Interfaces*, march 2007.

[169] R.-D. Vatavu. Interfaces that should feel right: Natural interaction with multimedia information. In M. Grgic, K. Delac, and M. Ghanbari, editors, *Recent Advances in Multimedia Signal Processing and Communications*, volume 231 of *Studies in Computational Intelligence*, pages 145–170. Springer Berlin / Heidelberg, 2009.

[170] R. D. Vatavu. User-defined gestures for free-hand tv control. In *Proceedings of the 10th European Conference on Interactive TV and Video*, EuroiTV '12, pages 45–48, New York, NY, USA, 2012. ACM.

[171] D. Vlasic, I. Baran, W. Matusik, and J. Popović. Articulated mesh animation from multi-view silhouettes. *ACM Trans. Graph.*, 27(3):97:1–97:9, Aug. 2008.

[172] D. Vogel and R. Balakrishnan. Interactive public ambient displays: transitioning from implicit to explicit, public to personal, interaction with multiple users. In *Proceedings of the 17th annual ACM symposium on User interface software and technology*, UIST '04, pages 137–146, New York, NY, USA, 2004. ACM.

[173] D. Vogel and R. Balakrishnan. Distant freehand pointing and clicking on very large, high resolution displays. In *Proceedings of the 18th annual ACM symposium on User interface software and technology*, UIST '05, pages 33–42, New York, NY, USA, 2005. ACM.

[174] A. von Kapri, T. Rick, and S. Feiner. Comparing steering-based travel techniques for search tasks in a cave. In *IEEE Virtual Reality Conference (VR)*, pages 91–94, 2011.

[175] J. Wang, S. Zhai, and J. Canny. Shrimp: solving collision and out of vocabulary problems in mobile predictive input with motion gesture. In *Proceedings of the*

SIGCHI Conference on Human Factors in Computing Systems, CHI '10, pages 15–24, New York, NY, USA, 2010. ACM.

[176] D. Weimer and S. K. Ganapathy. A synthetic visual environment with hand gesturing and voice input. *SIGCHI Bull.*, 20(SI):235–240, Mar. 1989.

[177] E. Westebring-van der Putten, W. Lysen, V. Henssen, N. Koopmans, R. Goossens, J. van den Dobbelsteen, J. Dankelman, and J. Jakimowcz. Tactile feedback exceeds visual feedback to display tissue slippage in a laparoscopic grasper. *Studies in health technology and informatics*, 142:420, 2009.

[178] A. Wexelblat. Research challenges in gesture: Open issues and unsolved problems. In I. Wachsmuth and M. Fröhlich, editors, *Gesture and Sign Language in Human-Computer Interaction*, volume 1371 of *Lecture Notes in Computer Science*, pages 1–11. Springer Berlin / Heidelberg, 1998. 10.1007/BFb0052984.

[179] D. Wigdor and D. Wixon. *Brave NUI World: Designing Natural User Interfaces for Touch and Gesture*. Morgan Kaufmann, 2011.

[180] A. Wilson. Using a depth camera as a touch sensor. In *Proceedings of ACM International Conference on Interactive Tabletops and Surfaces*, ITS 2010, pages 69–72. ACM, 2010.

[181] A. D. Wilson. Robust computer vision-based detection of pinching for one and two-handed gesture input. In *Proceedings of the 19th annual ACM symposium on User interface software and technology*, UIST '06, pages 255–258, New York, NY, USA, 2006. ACM.

[182] J. O. Wobbrock and K. Z. Gajos. Goal crossing with mice and trackballs for people with motor impairments: Performance, submovements, and design directions. *ACM Transactions on Accessible Computing (TACCESS)*, 1(1):4:1–4:37, May 2008.

[183] J. O. Wobbrock, B. A. Myers, and J. A. Kembel. Edgewrite: a stylus-based text entry method designed for high accuracy and stability of motion. In *Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology*, UIST '03, pages 61–70, New York, NY, USA, 2003. ACM.

[184] M. Wright, C.-J. Lin, E. O'Neill, D. Cosker, and P. Johnson. 3d gesture recognition: An evaluation of user and system performance. In K. Lyons, J. Hightower, and E. Huang, editors, *Pervasive Computing*, volume 6696 of *Lecture Notes in Computer Science*, pages 294–313. Springer Berlin Heidelberg, 2011.

[185] M. Wu and R. Balakrishnan. Multi-finger and whole hand gestural interaction techniques for multi-user tabletop displays. In *Proceedings of the 16th annual ACM symposium on User interface software and technology*, UIST '03, pages 193–202, New York, NY, USA, 2003. ACM.

[186] H. Wyss, R. Blach, and M. Bues. isith - intersection-based spatial interaction for two hands. In *2006 IEEE Symposium on 3D User Interfaces*, pages 59 – 61, march 2006.

[187] F. Yang, L. Ding, C. Yang, and X. Yuan. An algorithm for simulating human arm movement considering the comfort level. *Simulation Modelling Practice and Theory*, 13(5):437 – 449, 2005.

[188] X.-D. Yang, T. Grossman, P. Irani, and G. Fitzmaurice. Touchcuts and touchzoom: enhanced target selection for touch displays using finger proximity sensing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 2585–2594, New York, NY, USA, 2011. ACM.

[189] X.-D. Yang, T. Grossman, D. Wigdor, and G. Fitzmaurice. Magic finger: always-available input through finger instrumentation. In *Proceedings of the 25th annual ACM symposium on User interface software and technology*, UIST '12, pages 147–156, New York, NY, USA, 2012. ACM.

[190] K. Yatani, D. Gergle, and K. Truong. Investigating effects of visual and tactile feedback on spatial coordination in collaborative handheld systems. In *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work*, CSCW '12, pages 661–670, New York, NY, USA, 2012. ACM.

[191] K. Yatani, K. Partridge, M. Bern, and M. W. Newman. Escape: a target selection technique using visually-cued gestures. In *Proceedings of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, CHI '08, pages 285–294, New York, NY, USA, 2008. ACM.

[192] J. S. Yi, Y. ah Kang, J. Stasko, and J. Jacko. Toward a deeper understanding of the role of interaction in information visualization. *Visualization and Computer Graphics, IEEE Transactions on*, 13(6):1224–1231, 2007.

[193] M. Ying, A. P. Bonifas, N. Lu, Y. Su, R. Li, H. Cheng, A. Ameen, Y. Huang, and J. A. Rogers. Silicon nanomembranes for fingertip electronics. *Nanotechnology*, 23(34):344004, 2012.

[194] R. C. Zeleznik, K. P. Herndon, and J. F. Hughes. Sketch: an interface for sketching 3d scenes. In *ACM SIGGRAPH 2007 courses*, page 19. ACM, 2007.

[195] S. Zhai and P. O. Kristensson. Shorthand writing on stylus keyboard. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '03, pages 97–104. ACM, 2003.

[196] S. Zhai and P. O. Kristensson. Interlaced qwerty: accommodating ease of visual search and input flexibility in shape writing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, pages 593–596, New York, NY, USA, 2008. ACM.

[197] S. Zhao, M. Agrawala, and K. Hinckley. Zone and polygon menus: using relative position to increase the breadth of multi-stroke marking menus. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, CHI '06, pages 1077–1086, New York, NY, USA, 2006. ACM.

[198] S. Zhao and R. Balakrishnan. Simple vs. compound mark hierarchical marking menus. In *Proceedings of the 17th annual ACM symposium on User interface software and technology*, UIST '04, pages 33–42, New York, NY, USA, 2004. ACM.

[199] T. G. Zimmerman, J. Lanier, C. Blanchard, S. Bryson, and Y. Harvill. A hand gesture interface device. *SIGCHI Bull.*, 17(SI):189–192, May 1986.