Multimodal Interactions in Virtual Environments using Eye Tracking and Gesture Control

SHUJIE DENG

A thesis submitted in partial fulfilment of the requirements of Bournemouth University for the degree of

Doctor of Philosophy

June, 2017

Abstract

Multimodal interactions provide users with more natural ways to interact with virtual environments than using traditional input methods. An emerging approach is gaze modulated pointing, which enables users to perform virtual content selection and manipulation conveniently through the use of a combination of gaze and other hand control techniques/pointing devices, in this thesis, mid-air gestures. To establish a synergy between the two modalities and evaluate the affordance of this novel multimodal interaction technique, it is important to understand their behavioural patterns and relationship, as well as any possible perceptual conflicts and interactive ambiguities.

More specifically, evidence shows that eye movements lead hand movements but the question remains that whether the leading relationship is similar when interacting using a pointing device. Moreover, as gaze modulated pointing uses different sensors to track and detect user behaviours, its performance relies on users perception on the exact spatial mapping between the virtual space and the physical space. It raises an underexplored issue that whether gaze can introduce misalignment of the spatial mapping and lead to users misperception and interactive errors. Furthermore, the accuracy of eye tracking and mid-air gesture control are not comparable with the traditional pointing techniques (e.g., mouse) yet. This may cause pointing ambiguity when fine grainy interactions are required, such as selecting in a dense virtual scene where proximity and occlusion are prone to occur.

This thesis addresses these concerns through experimental studies and theoretical analysis that involve paradigm design, development of interactive prototypes, and user study for verification of assumptions, comparisons and evaluations. Substantial data sets were obtained and analysed from each experiment. The results conform to and extend previous empirical findings that gaze leads pointing devices movements in most cases both spatially and temporally. It is testified that gaze does introduce spatial misperception and three methods (Scaling, Magnet and Dual-gaze) were proposed and proved to be able to reduce the impact caused by this perceptual conflict where Magnet and Dual-gaze can deliver better performance than Scaling. In addition, a coarse-to-fine solution is proposed and evaluated to compensate the degradation introduced by eye tracking inaccuracy, which uses a gaze cone to detect ambiguity followed by a gaze probe for decluttering. The results show that this solution can enhance the interaction accuracy but requires a compromise on efficiency.

These findings can be used to inform a more robust multimodal interface design for interactions within virtual environments that are supported by both eye tracking and mid-air gesture control. This work also opens up a technical pathway for the design of future multimodal interaction techniques, which starts from a derivation from natural correlated behavioural patterns, and then considers whether the design of the interaction technique can maintain perceptual constancy and whether any ambiguity among the integrated modalities will be introduced.

Contents

List of Figures

List of Tables

List of Related Publications

- **Deng, S.**, Chang, J., Hu, S. and Zhang, J. J., 2017. Gaze Modulated Disambiguation Technique for Gesture Control in 3D Virtual Objects Selection. *3rd IEEE International Conference on Cybernetics (CYBCONF).* IEEE, 1-8. doi: 10.1109/CYBConf.2017.7985779

- **Deng, S.**, Jiang, N., Chang, J., Guo, S. and Zhang, J. J., 2017. Understanding the impact of multimodal interaction using gaze informed mid-air gesture control in 3D virtual objects manipulation. *International Journal of Human-Computer Studies*, 105, 68-80. doi: 10.1016/j.ijhcs.2017.04.002

- **Deng, S.**, Chang, J., Kirkby, J. A. and Zhang, J. J., 2016. Gaze-mouse coordinated movements and dependency with coordination demands in tracing. *Behaviour & Information Technology*, 35 (8), 665-679. doi: 10.1080/0144929X.2016.1181209

- **Deng, S.**, Kirkby, J. A., Chang, J. and Zhang, J. J., 2014. Multimodality with Eye tracking and Haptics: A New Horizon for Serious Games? *International Journal of Serious Games*, 1 (4), 17-34. doi: 10.17083/ijsg.v1i4.24

- **Deng, S.**, Chang, J. and Zhang, J. J., 2013. A Survey of Haptics in Serious Gaming. In: De Gloria, A. (ed.), *Games and Learning Alliance: Second International Conference, GALA 2013, Paris, France, October 23-25, 2013, Revised Selected Papers*. Cham: Springer International Publishing, 130-144. doi: 10.1007/978-3-319-12157-4_11

Acknowledgements

I would like to express my deepest gratitude to my supervisors Dr Jian Chang, Dr Julie Kirkby and Prof. Jian Jun Zhang for their guidance and support throughout the journey of my doctorate study.

Besides my supervisors, I would like to thank the examiners of my Viva Voce, Dr Jason Alexander and Dr Lihua You, for their insightful comments and hard questions which helped me refine the thesis.

I wish to extend my appreciation to Dr Nan Jiang, Dr Song-Hai Zhang and Prof. Shi-Min Hu for their invaluable support and advice on my study and research.

I would like to thank Ms Jan Lewis and Ms Sunny Choi for their support on my daily study at Bournemouth University. I am very grateful for the financial support approved by Prof. Iain MacRury and the Faculty Research Degrees Committee of FMC regarding my research and trips to conferences. I would like to acknowledge the AniNex research project (People Programme FP7/2007-2013/612627) for funding my secondment in Tsinghua University. I would also like to give special thanks to the Doctoral College of Bournemouth University for the studentship.

My sincere thanks also goes to the anonymous volunteers for their invaluable data and feedback of the experiments in this thesis.

I am lucky to have all the lovely colleagues and friends who shared Tolpuddle Annex 2 with me and I appreciate all the fun times we spent together.

Finally, I would like to thank my family and friends for their love and support.

This thesis is dedicated to my parents, Yibing Deng and Yanhui Cai.

Declaration

This thesis has been created by myself and has not been submitted in any previous application for any degree. The work in this thesis has been undertaken by myself except where otherwise stated.

The materials related to Chapter 2 have been partially published by Deng et al. (2013, 2014). The materials related to Chapter 3 have been published by Deng et al. (2016). The materials related to Chapter 4 have been published by Deng et al. (2017b). The materials related to Chapter 5 have been partially published by Deng et al. (2017a). Please find the full related publication list in Related Publications.

Chapter 1

Introduction

Figure 1.1: *A film still of Minority Report (2002)*

With the emergence of new technologies such as wearable devices, Virtual Reality (VR), Augmented Reality (AR), and Internet of Things (IoT), traditional mouse and keyboard can no longer satisfy the interactive requirements in these circumstances. Novel interaction techniques and interfaces beyond mouse are in demand for these technologies. A revolutionary trend for such user interface development is Natural User Interface (NUI) which allows users to interact with their natural behaviours as in sci-fi films (e.g., Figure 1.1). One way to realise natural interactions is multimodality that integrates multiple sensory modalities such as vision and touch into the interface and interaction design. However, multimodal integration may introduce challenges among each individual modality such as their spatio-temporal relationship, perceptual conflict, and ambiguities. Therefore, this thesis focuses on partially addressing these challenges for developing natural user interfaces and interactive techniques that relate to multimodal integration.

1.1 Background

User interfaces (UI) for human-computer interaction (HCI) provide access to interact with computers. The ten general principles for interaction design (Nielsen 1994) are: (1) visibility of system status; (2) match between system and the real world; (3) user control and freedom; (4) consistency and standards; (5) error prevention; (6) recognition rather than recall; (7) flexibility and efficiency of use; (8) aesthetic and minimalist design; (9) help users recognize, diagnose, and recover from errors; and (10) help and documentation.

UI has been developed through three stages over time as shown in Figure 1.2, Command line Interface (CLI), Graphical User Interface (GUI), and Natural User Interface (NUI).

CLI is a means for users to interact with computers by issuing lines of commands in the form of texts.

GUI is a type of interface that enables users to interact with computers through graphical components instead of typing commands. The typical paradigm for GUI is WIMP, i.e., windows, icons, menus and pointers. CLI and GUI utilise pointing, clicking and typing as the main input methods, i.e., mice and keyboards.

An NUI is a user interface designed to use natural human behaviours for interacting directly with content (Blake 2010). Apart from the general principles for all user interface design, an NUI should also consider the following four design guidelines (Blake 2011):

- *Instant expertise* indicates that an interface takes advantage of the user's existing skills, including domain-specific skills and common human skills.

- *Progressive learning* indicates that an interface provides a smooth learning path from basic to advance for users to learn how to use the interface.

- *Direct interaction* indicates that an interface is designed to enable high-frequency and contextual interactions.

- *Cognitive load* should be kept at a minimum by using innate abilities and simple skills in the interface design.

These guidelines of NUIs enable users to interact with computers using natural languages and behaviours such as gestures and gaze, even intuitive interactions which respond to ambient cues and intentional movements to create empathetic, personalised experiences (Kunkel et al. 2016). Novel interaction techniques are no longer privileges of sci-fi films or fantasies, they are actually affecting people's life and reshaping the future of industries and business.

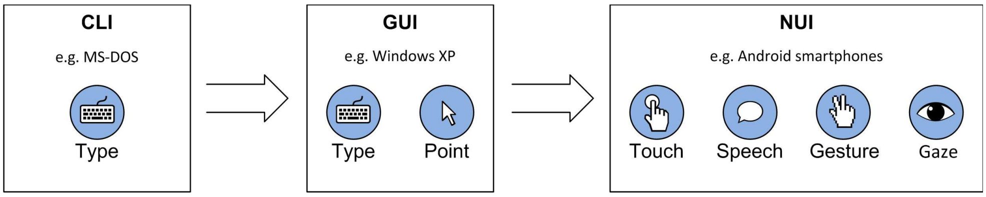


Figure 1.2: *Evolution of user interfaces.*

The surge of novel interaction techniques is usually related to new technology of how information is displayed. For example, mouse and keyboard are closely related to desktop or laptop displays. With the increasing popularity of portable and wearable displays such as smartphones, tablets and smart watches, touch gestures and speech have become pervasive in our daily life, representing that NUI has emerged to possess an important position in HCI which used to be dominated by mouse and keyboard. In recent years, even more cutting-edge technologies such as virtual reality (VR), augmented reality (AR), mixed reality (MR), and internet of things (IoT) provide brand new approaches for displaying information. The common feature of these new technologies is ubiquity that data are everywhere and information can be displayed anywhere. It encourages the emergence of immersive environment that blends both virtual and real worlds together. Innovate interface and interaction design are required in this circumstance to satisfy the new communication needs, which promises to accelerate the revolution of NUI.

According to Bryson (1996), VR/AR is the use of computer technology to create an interactive three-dimensional (3D) world in which the interactive objects have a strong sense of 3D spatial presence. Therefore, the two crucial elements in VR/AR are the affordance of 3D interaction and sense of spatial presence. The natural user interfaces suit the new VR/AR display because they intuitively enable both 3D interactions and sense of spatial presence.

More specifically, traditional CLI and GUI interfaces are designed for 2D interaction while VR/AR requires interacting within 3D space. Thus, new interface design needs to enable pointing, manipulation and more complex interactions in 3D space. For example, HTC Vive utilises two base stations to map a physical area with the virtual space to track the exact 3D locations of VR headset and hand-held controllers in the virtual space; Microsoft Kinect tracks the motion of our body parts using not only images but also depth information, so that we can use natural gestures to interact with the virtual space.

Furthermore, the sense of spatial presence is a result of immersive user experience which is highly related to the multimodality of interactions with virtual space. Multimodal interaction defines the way we employ natural modes of senses including vision, sound, touch, smell, taste and proprioception, both sequentially and in parallel, to passively and actively communicate with the external environment (Turk 2014). Human interactions with the physical world are inherently multimodal and we adapt this feature into HCI, specifically NUI, for gaining immersive user experience and sense of spatial presence in VR/AR. Accordingly, multimodal interfaces process two or more combined user input modes in a coordinated manner with multimedia system output (Oviatt 2012), such as speech, touch, gestures, gaze, head and body movements, as listed in Table 1.1.

Compared to unimodal interactions, multimodality plays an important role in user interface design by introducing the following advantages into interaction. Firstly of all, it permits a wider range of usage context for user interfaces. This context includes but not limited to users, tasks, and environmental conditions. This is because multimodality integrates more than one modality in interactions, which can either compensate the drawbacks among the modalities or correlate with all the modalities to enhance the performance of the interaction technique and user interface. Therefore, multimodal interactions can accommodate users either with or without handicaps; it can help relieve fatigue caused by overuse of one modality; it can prevent errors that easily occur when using only one modality; it can adapt to different tasks that suit interactions using a certain modality. Overall, multimodal interactions can support improved efficiency and precision of spatial information than a unimodal interface (Oviatt et al. 2000).

Furthermore, multimodality is proved to be able to enhance learning efficiency than unimodality. Moreno and Mayer (2007) suggested that multimodal virtual reality provided an attractive form of media to present learners with instructional materials in addition to words and pictures, which utilised the brain's capacity to process different information modalities through separate channels. This multimodal learning environment is highly interactive, combining with prior knowledge, it can promote deep cognitive processing in the learner. Guo and Guo (2005) further found that activation of unimodal prior knowledge, or unisensory memory retrieval, could be improved by multisensory learning conditions. They also found that cross-modal memory transfer could occur after preconditioning with bimodal stimuli followed by unimodal conditioning.

The state-of-the-art of human sensory modalities applied in HCI is listed in Table 1.1. This table is updated based on the list created by Blattner and Glinert (1996) which was firstly updated by Turk (2014). The examples listed in the table are typically components for building natural user interfaces, such as the intelligent assistant Siri (Apple Inc.) who deals with human speech.

Modality	Example
Visual	Gaze
	Facial recognition
	Emotion recognition
	Iris recognition
	Head movement
	Hand tracking and gesture recognition
Auditory	Speech
	Non-speech audio
Touch	Touch pointing, manipulation and gesture
	Tactile feedback (vibration etc.)
	Force feedback
	Kinaesthetic feedback
	Hand motion tracking
	Gesture
Other sensors	Sensor-based motion capture
	Electroencephalogram (EEG)
	Electromyography (EMG)
	Brain computer interface (BCI)

Table 1.1: *Human sensory modalities relevant to multimodal human-computer interaction updated from Blattner and Glinert (1996) and Turk (2014).*

Each sensory modality is a wide research area itself, and HCI has historically been focused on unimodal communication. Up to now, user interfaces have successfully integrated with many unilateral interaction modalities among which vision is the most dominant sense and touch has gained increasing attention for improving the natural user experience. I will introduce how each of both modalities is used in interactions and the current development of using both as multimodal interfaces in the next section.

1.2 Gaze + Gesture

In the last section I introduced the background of NUI, how NUI is evolved, why NUI suits VR context, and why multimodality is a key element for NUI design. In this section, I will discuss why combining gesture and gaze is a desirable multimodal interaction.

Vision as the most dominant sense has stimulated the intensive development of computer graphics technology that plays an important role in many visualisation applications. Such as in virtual simulation and VR/AR systems, the fidelity of visual feedback that is displayed either on a desktop or a headset greatly affects the system performance, and research on computer graphics helps improve it.

However, human eyes can not only be used as visual information receptors, but their movements can also inform interactions using eye tracking techniques. Eye tracking naturally reveals the region of interest by tracking users' gaze locations and their eye movement patterns, so it is widely used in cognitive science and psychology to decode human behaviour and attention. It is also started to be applied in interface and interaction design either as a direct input or a modulator to inform other inputs.

Apart from naturally corresponding to user's attention, eye gaze is capable of fast acquisition at the same time. It is reported that gaze can be $196ms$ faster than hand to reach the same target (Ariff et al. 2002). The existing gaze modulated techniques usually take advantage of it to enhance interactive efficiency.

Touch as one of the main human sense not only refers to the tactile perception, which helps us perceive temperature, texture, pressure, and shape of objects, it also represents proprioception (or kinaesthetic perception) which provides us with the capability to sense the movement and position of our limbs.

Proprioception combining with high Degree of Freedom (DoF) of our hands enables gesture control which facilitates natural interaction techniques and rich interactive expressions. It provides a transparent and natural way to interact using our bare hands as if there is no other medium between the virtual world and us. Besides, as discussed earlier gesture intuitively supports 3D interaction, so it is a more proper technique for AR/VR interactions. Note that there are gestures on a touchscreen (i.e., touch gestures) but also mid-air gestures which this thesis focuses on.

To combine the efficiency and naturalness, one may consider integrate gaze and gesture control as a multimodal interface. As aforementioned, the main purpose of multimodal integration is either to compensate drawbacks in one modality using others or to strengthen the interactive performance using the correlation among the multiple modalities. Consider that eye tracking can facilitate rapid target acquisition but lacks natural and expressive mechanisms to support modal actions (Chatterjee et al. 2015), while 3D hand gestures can afford the very action mechanisms that eye tracking lacks, combining the two modalities is expected to achieve rapid, expressive and natural 3D interaction experience.

Recent studies have already generated positive results using both eye tracking and gestures in interactions (Chatterjee et al. 2015; Velloso et al. 2015; Zhang et al. 2015). The basic paradigm is gaze modulated gesture control, which locates the object or region of interest using eye gaze and then enables manipulations of the object or interactions with the region using hand gestures. Current studies have investigated the performance of gaze combined with gestures by comparing it with gaze-only unimodal interaction techniques such as gaze dwell (i.e., stare for a certain time) and blink, and hand-only unimodal interaction techniques, such as mouse, trackpad and gestures. These studies all prove that combining gaze and gesture together can achieve better efficiency than the unimodal interaction techniques especially in target acquisition (Velloso et al. 2015), as well as reduced hand fatigue and increased ease of use (Zhang et al. 2015). However, these prototypes are still under study and it requires incrementally modification of the prototypes towards mature products to be used in real life.

The research findings have the largest impact on the real world user experience of gaze+gesture. It will assist the implementation in VR training software, for projects within different domains including medical, engineering, educational applications, where the integration of multiple sensory is vital. For instance, gaze+gesture has great potential to be applied in virtual surgery within VR context of use. It requires accurate and efficient 3D manipulation which gaze+gesture can provide.

In addition to the above serious usage, the research benefits future technological development within the gaming and interactive entertaining industry enormously by combining visual and touch information; clearly when achieved, this would enhance the user experience as well as hold the key to establish potential collaborations with enterprises to commercialise the developed techniques and software, bringing financial benefits to the stakeholders.

After discussing the advantages of gaze and gesture, positive results from existing research, and the potential impact on the wider society, it shows that combing gaze and gesture is potential to provide a synergy of efficiency and naturalness for interactions with AR/VR applications. This makes gaze + gesture a desirable multimodal interface to be studied in this thesis.

1.3 Motivation

Although current study has shown great potential of gaze modulated gesture control, this multimodal integration is still in its infancy, laying out many unanswered questions both in each individual modality and methods for multimodal integration.

Human eyes are always on, which brings in uncertainty of the meaning of eye movements. This is the Midas Touch problem (see more details in Section 2.1.2). With this problem, it is difficult to explicitly determine when the gaze variance should be effective. It also makes gaze modulated techniques sensitive to distractions and irrelevant eye movements such as blinks.

Although current eye trackers can deliver relatively accurate performance, it is still difficult for fine pointing. Most of the popular eye trackers use camera-based methods whose common issues may affect the accuracy of eye trackers, such as the sensitivity of environmental lighting, image noise, occlusion (e.g., eye lids), and abnormality (e.g., too large pupils). The physiological jitter and possible covert attention shifts of human eyes can also impact on the accuracy of eye tracking.

Another inconvenience introduced by the camera-based eye tracking techniques is the restriction of users' movements. As the performance of eye trackers is largely depending on the calibration accuracy, it is suggested users to limit their movement, especially head movement to the minimum. Therefore, recalibration is constantly required.

A common issue of mid-air gesture controls and vertical touchscreens is "gorilla arm" which indicates arm fatigue caused by holding our arm in front of our face for constant small movements of interactions without support to rest the arm.

Another drawback of mid-air gesture control that may degrade user experience of many serious applications, such as laparoscopic surgery simulation, is the lack of tactile feedback. This is a crucial feedback for more natural user experience because it is the feedback a user would expect when interacting in the real world. Besides, tactile feedback helps prevent interactive errors as more information is integrated and processed by our brain.

Similar to eye trackers, mid-air gesture control relies on camera-based methods for hand tracking and gesture recognition, so its accuracy is also affected by the common issues of camera-based methods. One particular problem is self occlusion, e.g., when an unfold palm is presented perpendicular to the camera, it is difficult to determine how many fingers are outstretched.

Some general issues exist in multimodal integration. First of all, the alignment of each individual modality is crucial for stronger multimodal integration because it is associated with the situation when the constituent unisensory stimuli arise from approximately the same location and at the same time (Stein and Meredith 1993). Therefore, understanding the spatio-temporal relationship among the multiple modalities is important for a better integration. In addition, it is vital to make sure each individual modality is well aligned with others, and to recover alignment when misalignment occurs.

An ambiguity occurs when more than one interpretation exist. It can be caused by multiple interpretations introduced by a single modality, or multiple interpretations introduced by different modalities. In addition, there is a phenomenon in psychology called “visual capture” indicating the dominance of vision over other sensory modalities. It may also cause ambiguity especially when misalignment occurs because users tend to believe what they see which may actually differs with what they perceive in other ways.

Another issues to be minded is to avoid cognitive overload because multiple inputs of data may easily introduce too much information. User interfaces are encouraged to be designed with a minimum cognitive load for the ease of learning and use, but introducing more modalities within a poor design may bring in more cognitive load which increases the difficulty in using such interfaces.

In terms of gaze+gesture, some issues may be highlighted based on the general issues discussed above. First of all, because of the importance of alignment of each modalities, it is recommended to find out how gaze and hand movement is aligned both spatially and temporally, and if there exists any misalignment such as conflicts between modalities, and cross-modal disparity.

Furthermore, gaze and gesture are both inadequate in pointing accuracy which cannot be simply compensated by each other, thus errors and ambiguities may occur using this technique. A solution to this problem is necessary as good accuracy is crucial in some cases where pointing error or ambiguity is intolerable, such as virtual surgery.

Moreover, there are many details in the design are still unclear, such as what triggers the shift from gaze pointing to gesture control, what are the gestures for different tasks, and how to deal with visual distractions. Each detail seems trivial but it may affect the final performance of the interface.

Overall, all the problems mentioned above may cause system instability and usability difficulties which will consequently degrade the user experience, so it is important to have them clarified.

In a nutshell, multimodal interactions combining gaze and gesture control have great potential to be applied in novel technology that requires natural user experience. However, its design requires further refinement to serve as a reliable user interface. Some of the problems are even becoming urgent as the area of VR/AR is making progress every single day.

This thesis intends to address some of the urgent problems including a perceptual conflict and an ambiguity problem in a timely manner. By addressing these problems, this thesis strives to contribute to the development of natural interaction design, provide theoretical and empirical insights for inspiring future integration of multimodal interactions, and benefit stakeholders of the engineering and entertainment industries.

1.4 Research Questions

Arising from the background and motivation statement is the research question of this thesis that how to integrate eye tracking and gesture control so that both efficiency and accuracy can be satisfied in a natural interaction environment.

Hence, the **AIM** of this thesis is to construct a theoretical basis and provide practical guidance on implementation of a multimodal interaction technique that integrates eye tracking and gesture control to facilitate natural human-computer interaction, which can be potentially applied in VR/AR.

Figure 1.3 illustrates the theoretical framework of this thesis, from which we can see that, to achieve the aim, the following three research questions need to be addressed:

Spatio-temporal relationship Eye tracking devices reveal the patterns of our eye movement and manual pointing devices are informed by our hand movement. As both devices are related to body movements, there must be a spatio-temporal relationship between the two, either they are irrelevant or they are correlated. It is important to understand how both modalities are aligned as discussed earlier. Eye-hand coordination is for describing this relationship. It is verified in physical space that our eye movement is positively

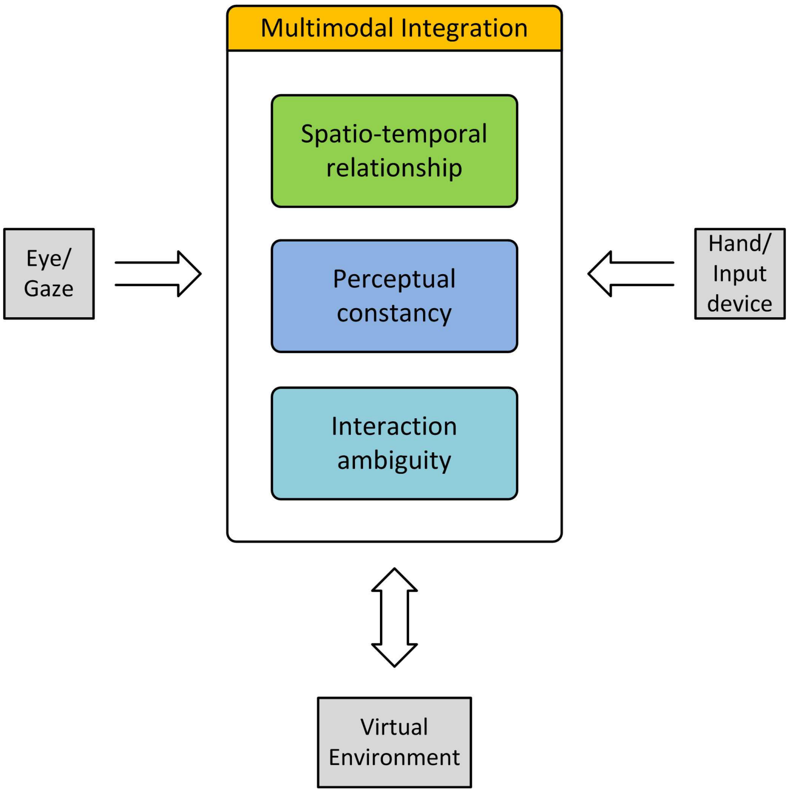


Figure 1.3: *Theoretical framework of this thesis.*

correlated with our physical hand movement in certain interactive scenarios. Specifically, eye movement leads hand movement. However, in human-computer interaction, eye movement and hand movement do not directly interact with the virtual space. They are interpreted into the virtual environment via various devices instead.

So the first research question arises that whether eye-hand coordination is still applicable in interactions that involve indirect pointing devices, such as a computer mouse. In other words, what is the spatio-temporal relationship between gaze and indirect input devices?

Perceptual constancy How we perceive a virtual environment concerns the relationship between physical properties of the virtual space (e.g., shape, size, colour, location, distance, or lighting) and our conscious experience of them. Perceptual constancy is for describing our tendency to perceive an object we are familiar with to constantly have the same physical properties even when the properties change (Pashler 2013). For example, an object may appear in different colours under different illumination but our brain can still recognise it as the same object. It is important to maintain perceptual constancy during interactions within the virtual space because it helps us to identify virtual contents when they appear or feel differently.

However, multimodal interaction introduces multiple inputs and each of them can change the physical properties of the virtual space. When a physical property is changed by one sensory modality but other modalities are not updated with the change, a perceptual conflict may be generated that users may misperceive this physical property because of the misalignment between the modalities. This conflict can confuse the users and affect their interactive decisions, and consequently impact on the quality of their user experience.

Hence, the second research question arises: is there any perceptual conflict in multimodal interactions that use eye tracking and gesture control?

Interaction ambiguity As discussed earlier, ambiguity may occur in multimodal integration. Disambiguation techniques or mechanisms are for solving the ambiguity problems. There are already studies about disambiguation techniques in gaze modulated interactions because eye tracking is not accurate enough to avoid pointing ambiguity. However, it remains unknown whether these disambiguation techniques can be adapted to gaze modulated gesture control. In the meantime, the disambiguation techniques should avoid introducing further ambiguity caused by gestures as gesture control is not accurate enough to remove ambiguities either.

Therefore, the third research question is: how can we build techniques to avoid the ambiguity problem in gaze modulated gesture control?

Accordingly, to address each of the questions, this work will mainly focus on the following objectives:

- Confirm that eye movement still correlates with and mainly leads hand movement in interactions that use indirect pointing devices (e.g., a mouse) so that using gaze movement to inform gesture control makes sense (Chapter 3).

- Examine the potential misperception problems that may occur during gaze modulated gesture control and propose their solutions (Chapter 4).
- Design and develop disambiguation techniques specifically for gaze modulated gesture control to avoid ambiguity introduced by either modality (Chapter 5).

1.5 Methodology

The research questions focuses on understanding the gaze+gesture technique in real world use and then providing design recommendations for improving its performance and naturalness, so the challenge of the methodology is what methods should be used to achieve this goal. Techniques such as behavioural modelling simulate the user as an information processing system with multimodal input and output components (Sebe 2009). These models can be used to evaluate the performance of a user interface but it requires prior knowledge or perhaps large amount of data to understand the design principles of the specific interface, and it is difficult to model users' subjective feedback. Therefore, this thesis uses the empirical method which actually builds the interaction techniques, and then designs controlled experiment and collects users' feedback to evaluate the usability of the techniques.

To solve the first research question, both gaze movements and mouse movements are recorded in a controlled tracing experiment. Using the observed data, the linearity between both modalities are examined in all conditions. The temporal difference and spatial disparity are quantified.

To solve the second research question, a problem is observed from real world use and a definition of the problem is developed from it. Controlled experiments are conducted to test the definition and three solutions are compared both quantitatively and qualitatively.

To solve the third research question, a disambiguation technique is developed and compared with traditional mouse pointing in controlled experiments both quantitatively and qualitatively.

Empirical studies are not a formal proof of a fact. They rather yield, support, or reject hypotheses (Weibelzahl and Weber 2002). However, the results are always afflicted with uncertainty, which can often be expressed in a statistical probability value, and the confidence intervals, test power, and effect sizes are available. In this thesis, all results of the controlled experiments are statistically evaluated and properly reported.

1.6 Contribution

Based on the aim and objectives mentioned earlier, the contributions of this thesis include:

1. An empirical study confirms gaze linearly correlates with mouse cursor movement and quantifies gaze lead time in continuous manipulation, which broadens the applicability of the existing eye-hand coordination theory to human-computer interactions that use indirect input devices regardless of the type of the task, i.e., discrete or continuous tasks.

2. Experimental findings are reported that gaze lead is positively correlated with gaze-mouse coordination demands, and leading distance is related to the trace shape but it is constantly within a visual acuity range. These findings can serve as guidance for further designs of interaction techniques that involve gaze and indirect hand control coordination.

3. A method of calculating the time difference between the correlated eye movement and hand movement is proposed. It can avoid directional disparity, which provides an alternative approach when more than one dimensional movement is involved and the concern is the overall time difference instead of a time difference of each dimension.

4. Data of the experiment that records gaze and mouse movements is available online for the convenience of other researchers of their related studies. [Link: https://github.com/blackdeng/gaze_mouse]

5. A multimodal disparity issue that causes perceptual conflict, the spatial misperception problem (please see Chapter 4 for the definition), is defined. To the author's best knowledge, it is the first time that the cause of interactive interruptions that introduced by gaze modulated pointing is specified.

6. Strategies and resolutions for the spatial misperception problem are proposed, whose comparative usability and users' preference are further investigated. It contributes to enriching the design guidance for future implementations involving gaze modulated gesture control.

7. A novel disambiguation technique is proposed specifically for consummating the design of multimodal interactions that involve gaze and gesture control.

8. A comparative user study with the conventional pointing techniques helps us further understand the usability of the proposed disambiguation technique, which directs us for more robust interaction design.

Overall, This study looks into the spatio-temporal relationship between gaze and hand motion, identifies and solves possible perceptual conflicts and interactive ambiguities, to bring a synergy of novel technology with the natural and transparent user experience. For the publications relating to this study, please refer to Related Publications.

1.7 Thesis Structure

This chapter describes the motivation of the development of multimodal interaction combining eye tracking and gesture control in virtual environments, specifies the potential risks and problems, and highlights the contribution of this thesis. The rest of this thesis is structured as below:

Chapter 2 sums up the related work of this thesis. It firstly provides an overview of unimodal interactions using eye tracking, and then hand control, respectively. The current state-of-the-art in multimodal interactions combining the both modalities is then presented starting with the behavioural pattern of eye-hand coordination followed by the existing multimodal paradigms combining eye tracking with different hand control devices. I finish the related work by a brief review of the multimodal challenges, especially the proximity and occlusion problems that may still occur in gaze modulated multimodal interactions, together with their existing solutions.

Chapter 3 investigates the spatio-temporal relationship between gaze and mouse cursor movement in 2D tracing tasks under three different levels of coordination demand. This chapter specifically elaborates the details of how the experiment was designed and the definition of the quantitative measurements for data analysis. The experiment results are further discussed in terms of the behavioural pattern, linearity, time and distance difference between gaze and mouse cursor movement.

Chapter 4 defines the spatial misperception problem and proposes its possible solutions. It firstly reviews under what background and conditions this problem may occur by comparing existing gaze modulated techniques. Following that, the problem is defined and then the strategies to tackle the problem are discussed. Three methods to resolve this problem according to the strategies are proposed. The experiment is designed to validate the problem definition and compare the usability of the three proposed methods. The results confirm the definition and reveal insights of the three proposed methods in terms of their advantages and disadvantages.

Chapter 5 proposes a new disambiguation technique for gaze modulated gesture control, which follows the coarse-to-fine two-step concept. This technique firstly uses a cone for ambiguity detection, then a gaze probe to declutter the ambiguous objects. A state transaction of a drag-and-drop task using this proposed technique is then given. A comparative experiment using the drag-and-drop task was conducted for a better understanding of this technique.

Chapter 6 concludes the thesis with a summary of the findings, recommendations, implications, and discussions about future work.

Chapter 2

Related Work

Unimodal interactions using only gaze or only hand control are well studied respectively. With increasing interests of the concept of NUI in recent years, the integration of these two modalities has emerged. It is based on the psychological findings of eye-hand coordination with the purpose of taking advantage of one modality to compensate the drawbacks of the other and utilising their correlation to strengthen the interactive performance. However, there still exist many unsolved problems in this multimodal integration, two of which that may greatly impact user performance are pointing inaccuracy and occlusion because they can cause selection ambiguities.

Therefore, in this chapter, I review the two types of unimodal interactions, gaze and hand control, followed by the current state-of-the-art multimodal interaction techniques combining these two. After that, I further review the existing studies for resolving the pointing inaccuracy and occlusion problems in 3D spatial interactions.

2.1 Eye Tracking in Unimodal Interaction

Our eyes do not operate as we might imagine, smoothly scanning the visual environment like a video camera and erratically encoding information. Rather vision is a highly active process, which is partially due to neural processing limitations of the retina, i.e., the visual resolution declines rapidly extending from the fovea on the retina, see Figure 2.1. Therefore, the brain continuously analyses the visual environment and selects the most salient aspects to be further processed. The selection and processing are typically realised by two physiological eye movements, saccades and fixations.

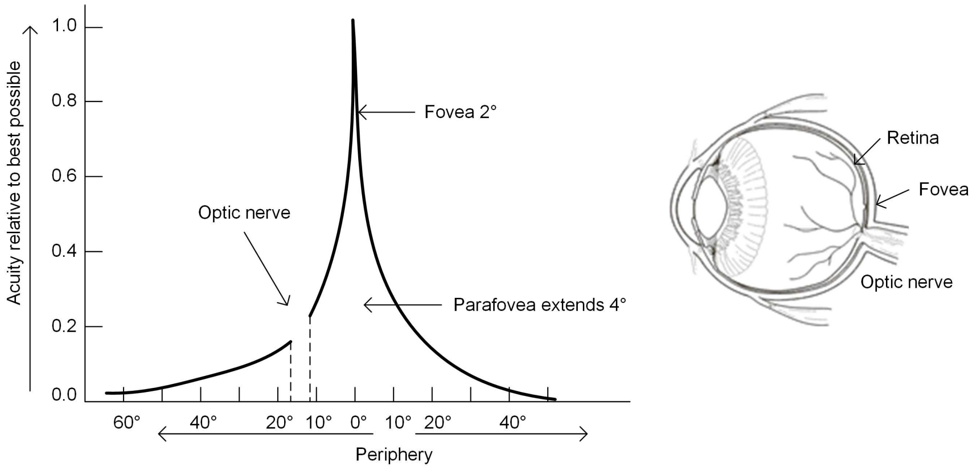


Figure 2.1: *Representation of the rapid resolution decline of the retina extending from the fovea.*

Our eyes typically make 3-4 saccadic movements per second, which last only a few hundredth of a second each. Saccades enable us to align our fovea, or the high acuity part of the retina, on the most informative aspect of the scene; with this type of scanning behaviour we render ourselves virtually blind for considerable periods of time, as during a saccade we experience what is known as saccadic suppression, where no new information is taken in (Liversedge and Findlay 2000).

In between these scanning movements, there are times where our eyes are relatively still, and during these periods visual information is encoded and processed. In fact, during many tasks, the eyes remain fixated until the stimulus is fully processed. These periods are called fixations, as such the time course of a fixation is an important indicator of visual processing during a task.

Other than typical fixations and saccades, basic eye movements also include smooth pursuit, vergence, and vestibulo-ocular movements (Purves et al. 2001). Smooth pursuit is the movement that the eye makes as it tracks a moving stimulus. It is a voluntary movement which is slower than the saccades. Most people are not capable of smooth pursuit in the absence of a moving target. Vergence is the only disconjugate movement of both eyes that they either converge towards each other to look at closer targets or diverge away from each other to look at targets further away. Vestibulo-ocular movements compensate the head movements by constantly adjusting the eyes position to the opposite direction of the head movements to stabilise the image on the retina (Purves et al. 2001).

Pupil size is another important measurement which can indicate user's task engagement. The diameter of a pupil is considered to reflect cognitive load, i.e. when the pupil dilates, it indicates increased cognitive processes occurring in the brain (Granholm and Steinhauer 2004). For example, it was applied in measuring cognitive load in driving simulators (Palinko et al. 2010). Moreover, as the pupil acts as the aperture of the eyes, it can influence the depth of focus. Typically, the pupil dilates when the depth of focus decreases (Reichelt et al. 2010). Alt et al. (2014) took advantage of this feature to determine the depth of user's gaze in 3D stereoscopic displays. However, pupil dilation is not always considered a reliable indicator of learning (Schultheis and Jameson 2004) and it can be easily disturbed by the varying lighting conditions in the interaction environment.

Although there is a long history of observing eye movements within psychology and related fields, see Rayner (1998, 2009) for reviews, it is only recently that researchers have begun to introduce eye tracking as an input modality in human-computer interaction. As such, over the past 20 years, there have been considerable interests in studies that interact using the aforementioned eye movements either implicitly or explicitly. The implicit usage analyses the eye movement data and applies the learned data patterns to benefit interaction. The explicit usage directly inputs gaze position into real-time interaction.

Analytical tool Offline or online analysis of eye movement behaviour is often utilised in order to understand the user's performance in human-computer interaction (Poole and Ball 2006; Ehmke and Wilson 2007). Eye movements demonstrate the user's responses to visual changes in the virtual environments as well as behaviours the user undertake during visual search tasks, with this data we are able to extrapolate how users engage in the learning and interaction process. Issues like the fixation duration, saccade length, size of the perceptual span (the functional field of view), as well as, where and when viewers move their eyes during these tasks are dynamic means by which we can assess the usability of the interactive techniques. For example, patterns of eye movement behaviour provide a measure of task difficulty and user engagement. Vlaskamp and Hooge (2006) reported that the number of fixations and fixation duration increased and saccade amplitude decreased with increasing crowding in the visual search tasks.

Interaction inputs Conventionally, users have primarily been able to interact with the virtual environment using hand motion. However, eye movement can now act directly as a cursor in the virtual environment replacing the traditional keyboard and mouse inputs. It can be used for basic interactive tasks such as pointing, navigating, and level of detail (LOD) rendering by way of gaze contingent paradigms (Reingold et al. 2003; Duchowski et al. 2004; Duchowski 2002). These interactions are enforced by eye gaze - the overt attentional position of the user. Although attention is not always represented by gaze because of covert orienting (Posner 1980), it is assumed that in an intensively engaging scenario such as a videogame, gaze reflects the region of interest (Sundstedt et al. 2008).

2.1.1 Eye tracking devices

To either analyse the eye movements or directly input the gaze position in interactions requires data of the eye movements and gaze position. The eye trackers are used for obtaining the eye positions, gaze positions, and eye movements. Note that the eye position is different to the gaze position. The eye position is the position of the physical eye relative to the display while the gaze position is a 2D position on the display indicating where the user is looking. The temporal trajectory of the 2D gaze position reflects the spatio-temporal movement of the eyes in the real 3D world. By analysing the eye or gaze movement trajectory patterns, computers can identify the types of the eye movement, i.e., saccades, fixations or blinks.

Generally, there are three types of eye trackers depending on how they work, electrooculography (EOG), search coil, and camera-based tracker. In EOG, two pairs of electrodes are placed around the eyes, typically above and under, and to the left and right of the eye, to obtain eye positions according to the signals between each pair of electrodes (e.g. Figure 2.2 (a)). The electrodes measure the electrical potentials between the cornea and the retina. The potential signals provide an indirect availability to interpret the eye positions and gaze positions. It is sensitive to external noises thus it has limitations in accuracy and precision.

The search coil has embedded inductive coil wires in a contact lens, which is surrounded by several magnets (e.g. Figure 2.2 (b)). The eye movement changes the movement of the coil so the generated electric currents change accordingly, by measuring which the eye position can be obtained. This method is also depending on the stability of the signals. Moreover, it is intrusive to the user's eyes which can degrade the interactive comfortableness.

The camera-based eye trackers capture the image of the user's eye or eyes, and determine the eye positions according to the pupil and light reflection positions on the image (e.g. Figure 2.2 (c) and (d)). The camera uses infrared lights to obtain clearer images to highlight the details of the eyes. Normal cameras are also used in basic eye-based interactions such as a mobile phone. Camera-based eye trackers can be remotely placed with the display, or the users can wear them as glasses. They are unobtrusive and easy to apply compared to the other two types, which makes them the most common type among various eye trackers. Typically, to enable eye tracking within VR applications, it requires eye tracking enabled VR headsets.

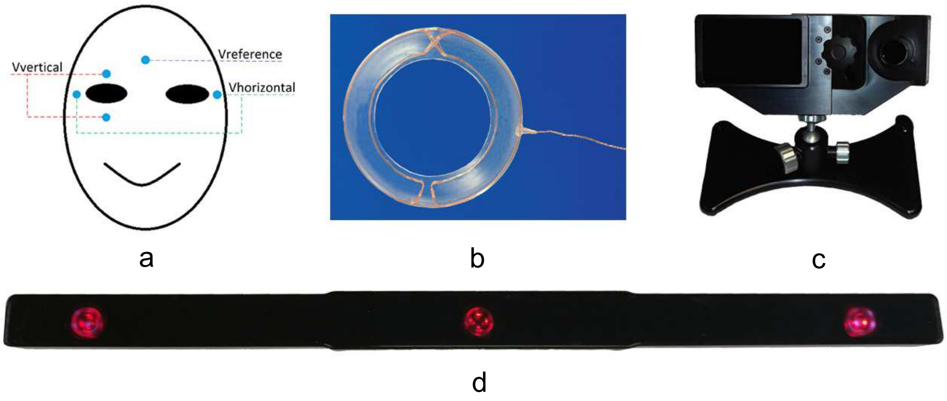


Figure 2.2: *Different types of eye trackers. (a) A typical arrangement of EOG electrodes to record horizontal and vertical eye movements. The blue dots represent the electrodes. The reference electrode is in the centre of the forehead. (b) 3D Scleral Search Coil (Chronos Vision GmbH). It measures the horizontal, vertical and torsional eye positions. (c) EyeLink 1000 (SR Research Ltd.). This is a camera-based eye tracker to be used with a chin rest. (d) Tobii EyeX. This is a camera-based eye tracker for games.*

Eye tracking devices have great variability in terms of their spatial and temporal frequency. Eye trackers for academic purposes usually have higher sampling frequencies (e.g. $1000Hz$) to capture the most subtle eye movements. Thus, in the remote camera-based eye trackers, the head movement should be limited to the minimum. For example, a chin rest can be used for this purpose. The EyeLink 1000 as shown in Figure 2.2 (c) is usually used in this case. Eye tracking for consumers are designed for daily use, so comfortableness in continuous usage and movement flexibility should be considered as a trade-off with the accuracy. They are more affordable compared to the high accuracy eye trackers, and their lower sampling frequency (e.g. $60Hz$) limits the tracking of only fixation behaviours, because the ballistic saccadic movements that typically only span tens of milliseconds are beyond their capability. The Tobii EyeX as shown in Figure 2.2 (d) is a typical example of this type of eye trackers.

Considering the aforementioned advantages and disadvantages of each eye tracking device, I choose camera-based eye trackers in this thesis because the interactive usability is my priority and camera-based eye trackers are far more comfortable than the EOG and search coils. More specifically, in the experiment in Chapter 3, I use the high accuracy eye tracker EyeLink 1000, because this experiment is a quantitative study of human behaviour which demands a high quality of the measurement data. In the interaction technique prototypes implemented in Chapter 4 and 5, I use the game device Tobii EyeX, because such cost-effective devices are designed for real life. The details and specifications of the eye trackers used in each experiment are elaborated in each chapter respectively.

2.1.2 Applications

Typically, state-of-the-art interactive application of gaze behaviour can be divided into two categories, one involves voluntary gaze control and the other uses reflexive gaze movements. Voluntary gaze control represents the purposeful direction of gaze in order to scan more precisely a specific region of interest and is used primarily in pointing and gaze gestures for example. Duchowski (2002) has summarised the uses of online real-time recording of voluntary gaze control as text scrolling, activating game character behaviour, accessing a virtual keyboard, and accelerating cursor movements. Whereas the reflexive gaze movements signal more automatic attention allocation to a particular region of interest and are used primarily for egocentric camera navigation and updating LOD rendering.

Using reflexive gaze movements, egocentric camera control provides spontaneous view changes but also guides user's visual attention by element composition. Hillaire et al. (2008) implemented a method of first-person camera navigation using eye tracking. Burelli and Yannakakis (2011) developed an artificial neural network (ANN) camera behaviour prediction model by analysing eye tracking data collected from a game. It achieved over 70% accuracy for different types of game action. Similar applications have been extended to teleoperation. Zhu et al. (2011) implemented a gaze-driven remote camera control with the straightforward principle that it moved the region of interest into the centre of the screen.

LOD rendering is a technique where geometric objects are represented at a number of resolutions and the most appropriate one will be selected at any point based on the workload of a graphics system (Reddy 1998). It is suitable for applications with complex simulation but require real-time response, such as surgical training, which always incorporates a large amount of fine meshed deformable tissue that is computationally expensive. There are two types of gaze contingent display used to implement LOD rendering. One is the screen-based display, which manipulates pixels and matches the graphical display with vision mechanisms. With this approach, it assigns higher resolution to the fixation vicinity and a lower resolution to peripheral areas. The other is the model-based display, which statically or dynamically computes fine-to-coarse meshes of an object. Fine structure is rendered when gazed upon or coarse structure when gaze recedes (Duchowski 2002). Hybrid methods featuring local connectivity and rendering efficiency have been proposed. Murphy et al. (2009) used Contrast Sensitivity Function (CSF) and ray casting in order to build a hybrid method, where CSF was utilised to describe the amount of visible detail changes conforming to gaze contingency and ray casting to avoid direct manipulation of the mesh.

Most eye-based interactions depend on the fixations of the eye movement, as I just described. There are also applications using saccades, smooth pursuit, and vergence to modulate interactions. Typically, these eye movements are mostly reflexive as for instance, it is impossible for the users to adjust their pupil size by themselves.

Saccadic suppression is applied in saccade contingent updating to prevent the users from noticing sudden display difference. Saccade contingent update can separate “what you see” and “what you see next” by changing the peripheral scene that is outside of foveal vision (Kawashima et al. 2005). It is also used to hide graphic updates if the update happens within a saccade. Compared to a smooth scene change, this method enables immediate large scene change with no disturbance to the users because the change is not detectable during saccades (Schumacher et al. 2004). In applications such as videogames and teleoperation, real-time performance is important, so the latency of detecting a saccade needs to be as small as possible for a seamless experience. New saccade detection methods have been proposed for reducing latency, therefore, leading to the possibility of achieving real-time performance and enhanced visual experience (Franke et al. 2014; Watanabe et al. 2007, 2012). Furthermore, using saccadic behaviour to predict fixation location can provide the capacity for seamless implementation of LOD graphic rendering (Triesch et al. 2002), as previously applied in high-performance flight simulators.

Interactions using smooth pursuits recently emerged. Vidal et al. (2013) designed a pursuit-based interaction technique that enabled spontaneous interaction with no need of eye tracking calibrations. This technique correlates the smooth pursuit movement of the eyes with the trajectory of the moving object on the display they are following to avoid obtaining the absolute gaze positions. However, it can still be used in calibration, which is more implicit and robust than the traditional marker viewing calibration methods (Pfeuffer et al. 2013). It was further applied to overcome selection ambiguity introduced by the eye tracking inaccuracy (Velloso et al. 2016).

Binocular coordination of the two eyes especially the vergence movement is useful in 3D stereoscopic displays because it can inform the 3D depth of the gaze position. Pfeiffer et al. (2008) has investigated its usability in VR environments. Alt et al. (2014) tested the depth accessibility using the vergence in 3D target selection tasks. Templin et al. (2014) obtained the eye tracking data from a user experiment that measured the vergence response time to stereoscopic depth change and derived a prediction model to estimate the time users need to adapt to a new 3D scene with a different vergence. Sudden temporal depth changes can happen a lot in 3D film scene transitions and an estimation of the response time can help the editors optimise the video editing.

Midas Touch Using eye tracking in a unimodal interaction may have the Midas Touch problem which defines the situation when an unintentional action is triggered accidentally by eye movements. Naturally, the eyes are for observation. It has the second role in issuing control commands if it has been designed to do so. In this case, an observing gaze may be misunderstood as a trigger of an action, because gaze commands are not as explicit as a mouse click. Thus, eye tracking data is noisy as not every eye movement is intentional. Dwelling fixations and blinks were proposed to use as a confirmation (Jacob 1990). However, their performances were not comparable with the traditional mouse interactions in selection tasks (Chatterjee et al. 2015).

2.2 Hand Control in Unimodal Interaction

In reality, we use our bare hands or hand-held tools as the main approaches to naturally interact with the environment surrounding us. In virtual environments, various hand-controlled techniques and devices provide different input methods to map human behaviours from the physical space to the virtual space. Depending on the different mapping techniques, the hand-controlled techniques can be categorised into absolute and relative pointing techniques; depending on whether an intermediary exists between the input and the output, the techniques can be categorised into direct and indirect input techniques.

Absolute and relative pointing In absolute pointing, a point in the input space is consistently mapped to a point in the output space, so it is a point-to-point consistent mapping between the input and the output. Touch screen, stylus-based haptic devices and some camera-based sensors are absolute pointing devices, i.e., the pairing between the hand position and its mapped position in the virtual space is fixed.

Relative pointing techniques map displacements instead of points from the input space to the output space, so they do not require a fixed point-to-point mapping. Mice, joysticks and touchpads are typical relative pointing devices that the relative position of the hand and the virtual cursor are not fixed. Once a user lifts the device (e.g., a mouse) or their hand, the mapping will be re-calibrated.

The relative pointing techniques do not strictly require a fixed Control-Display (CD) gain which, however, is essential to the absolute pointing techniques. Note the CD gain is a function of the velocity that reflects the ratio between the control device and the display pointer movements. When the CD gain is greater than 1, the control device moves faster than the pointer and vice versa (Casiez et al. 2008).

Direct and indirect input Devices using direct input techniques enter body movement data directly to the system, so the body movement and the input data are equivalent. It does not require conscious mental translation (McLaughlin et al. 2009). A touch screen is a typical direct input device which is pervasively adopted to most smartphones and tablets nowadays.

Indirect devices do not input body movement into the system equivalently as what the direct input devices do. Instead, a transformation from the body coordinates to the system coordinates is introduced. A virtual cursor is usually used for providing visual feedback of the region of interaction. For example, the hand moves the mouse on a horizontal desktop but the movement is translated into the cursor movement on a vertical screen. Moreover, the CD gain can contribute to the transformation because a large CD gain can result in a large movement of the device that only corresponds to a small movement of the cursor and vice versa. All the components that are involved in the transformation are integrated into our brain to build a mental representation of the transformation. Depth sensors, mice, graphics tablets, and joysticks are typical indirect input devices.

Direct input provides stronger affordance in selection (Shneiderman 1991). However, indirect input enables manipulation on a distant and large display (Vogel and Balakrishnan 2005; Ballagas et al. 2006). It also alleviates the hand occlusion problem (Forlines and Balakrishnan 2008) that the user's hand covers portions of the display.

2.2.1 3D spatial interaction

According to Poupyrev and Ichikawa's taxonomy of virtual environment manipulation techniques (Poupyrev and Ichikawa 1999), if the users interact with the virtual environments from outside, i.e., the God's viewpoint, it is exocentric interaction, such as the World-In-Miniature metaphor (Stoakley et al. 1995; Andujar et al. 2010). Otherwise, if the users interact from inside of the environment, it is egocentric interaction which is the most common interactive approach in VR. In egocentric interaction, there are two basic metaphors, virtual pointer and virtual hand. A virtual pointer is used to select by emitting a vector. If the vector intersects with the target, it can be picked up and manipulated. The virtual hand metaphor utilises a 3D cursor, sometimes in a human hand shape, whose movements corresponds to a tracked physical hand movement. When the virtual hand collides with a target and a selection confirmation is triggered, the target will be attached to the virtual hand and manipulation is enabled (Poupyrev and Ichikawa 1999).

Mice and touchscreens are 2-DoF devices which enable direct pointing on a planar surface. Their interactions in 3D space are based on the virtual pointer metaphor with the help of ray-casting techniques. Camera-based depth sensors and haptic devices provide 3- or 6-DoF, including positional and rotational configuration on all three dimensions, so their interactions are based on the virtual hand metaphor.

Ray-casting Using the 2-DoF devices to interact with the 3D virtual environment usually requires additional assistance to obtain the depth information. Ray-casting is widely used for this purpose which shoots a ray, either visible or invisible, from a point into the 3D space. The first object who is penetrated by the ray is selected as the target (Mine 1995). There are three parameters of ray-casting can be configured to vary this interaction technique, the point of origin, the direction and the shape of the ray.

The point of origin is typically controlled by the user's hand/finger, or hand-controlled virtual cursor because the pointing devices are mostly hand-held or they tracks the hand movements. However, there are also ray-casting originated at the user's eye (Argelaguet et al. 2008; Tanriverdi and Jacob 2000), head or chin (Hincapié-Ramos et al. 2015).

The point of origin combining with the ray direction enables more flexibility in ray-casting control. A 2-DoF pointing device is not competent to adjust the ray direction as it can only provide a 2D dot for pointing, so its ray direction can be considered as constantly perpendicular to the camera's clipping plane. However, with more DoFs, such as a wand who can rotate about the depth axis (Grossman and Balakrishnan 2006; Hincapié-Ramos et al. 2015), a pointing device is able to configure the ray direction.

When an eye is the point of origin, users can adjust the ray direction by their hand. For example, in the Sticky Finger technique (Pierce et al. 1997), a user can point to a virtual object using a single outstretched finger, and a ray is cast from the user's eye point through the outstretched fingertip into the scene. Obviously, the ray direction can be adjusted by fingertip manoeuvre. Argelaguet and Andujar (2009) proposed another technique that indirectly controls the ray direction by hand rotation.

The shape of the ray can be varied, such as a cone (Steed and Parker 2004; Steed 2006). These volumetric variations are mainly for solving the selection ambiguity problems which I will discuss in detail in Section 2.4.

Sometimes a ray can penetrate several objects and it is not always the first object who is expected to be selected. In this case, the flexibility in depth movement is required. Feiner and Steven (2003) designed a technique using curved rays to bypass obstacles to enable pointing to the occluded objects. Wyss et al. (2006) proposed the iSith technique using two rays originated from both hands so a 3D intersection point could be determined in different depths. Argelaguet and Andujar (2009) also utilises two rays but one is originated from user's eye. Grossman and Balakrishnan (2006) elaborated four disambiguation techniques including depth ray, lock ray, flower ray, and smart ray. Specifically, the depth ray and lock ray techniques integrate a depth marker attached with the ray so moving the input device forwards and backwards can enable the marker to intersect with objects of different depths.

Virtual hand The virtual hand metaphor provides intrinsic affordance of all three dimensions, in which hand movements are typically virtualised as a virtual hand or a 3D cursor. Although this advantage enables a more natural interactive experience, solving the following issues made it a more robust interaction technique.

First of all, it can be difficult to reach objects outside the arm reach when the physical space and virtual space are identically mapped, i.e., the CD gain is 1. Poupyrev et al. (1996) proposed the Go-go technique to enable distant selection by adjusting the CD gain. When the distance between a user's hand and their torso is beyond a threshold, the CD gain is changed so the virtual hand is elongated non-linearly to reach the distant object.

Additionally, virtual hand is not as accurate as the virtual pointer metaphor due to two reasons, the physiological trembling and jitter of human hands, and a requirement of higher depth perception as the hand's depth is used to control the virtual hand/cursor. Therefore, disambiguation mechanisms are required (see details in Section 2.4).

For more details of 3D interaction techniques, please refer to the survey by Argelaguet and Andujar (2013).

2.3 Gaze Modulated Multimodal Interaction

Combining eye tracking and hand-controlled interaction techniques, the gaze modulated manual inputs are designed based on the eye-hand coordination theory, in which the gaze enables natural selective attention and fast target acquisition while the manual input enables fine grainy pointing or natural rich expressions depending on the feature of the integrated hand-controlled device, including mouse, touch screen, gesture control, and haptics.

2.3.1 Eye-hand coordination

Eye-hand coordination describes the coherent control of eye movements and hand movements with visual input as well as proprioceptive feedback. It is task specific so it has been studied in various human behavioural tasks, including object reaching and pointing (Biguer et al. 1982; Neggers and Bekkering 2000; Ariff et al. 2002; Crawford et al. 2004; Masia et al. 2009), web browsing (Chen et al. 2001; Rodden et al. 2008; Guo and Agichtein 2010), goal-directed aiming (Binsted et al. 2001; Behan and Wilson 2008), visually guided tracking (Gauthier et al. 1988; Vercher and Gauthier 1992; Xia and Barnes 1999; Tramper and Gielen 2011), drawing (Reina and Schwartz 2003; Gowen and Miall 2006; Coen-Cagli et al. 2009; Tchalenko and Miall 2009), and trajectory tracing (Gowen and Miall 2006; Tramper and Gielen 2011). Other more complex tasks or applications that combine sequential movements have also attracted growing research interests, such as object manipulation (Johansson et al. 2001; Bowman et al. 2009) and virtual laparoscopic surgery (Yamaguchi et al. 2007).

Although there are many different tasks, they can roughly be divided into two categories, discrete actions and continuous manipulation (Chatterjee et al. 2015).

- Discrete action is a single hand motion, such as pinching, swiping and grabbing. It is commonly associated with giving a command or sending a confirmation as a trigger, for example, pressing a button to open a window or swiping to read the next page.
- Continuous manipulation involves constant positional changes in three dimensions, such as dragging and tracing.

These investigations into physical visuomotor tasks have revealed evidence of spatiotemporal leading of gaze position to hand movement with a high correlation between the two. The eye movement is typically represented by the corresponding gaze position, so we describe using gaze positions for the sake of accuracy. Johansson et al. (2001) designed an object reach and grasp task, and found that gaze provided visual guidance for hand movement by marking the critical position where the fingers were reaching or targeting the object. In a curve drawing task, Reina and Schwartz (2003) noted that gaze position clustered into several groups along the trajectory of the hand movement; they found that gaze remained still while the hand was approaching the object or moving away from it, then the gaze saccaded ahead of the hand position onto the next cluster. The fixation clusters tended to be located near high curvature areas along the hand trajectory, and the saccades occurred when tangential hand velocity reached a local minimum.

Various methods for quantifying the lead time have been developed based on the complexity of the tasks. In the discrete tasks such as point-to-point tasks, goal-directed reaching, pointing, or tapping, they normally require a single saccade or two. By calculating the difference in time between gaze on target and hand on target discretely, the lead time of gaze can be straightforwardly quantified. Ariff et al. (2002) designed a pointing task with unseen stimuli during hand movements on a horizontal plane. They reported that the saccades constantly occurred at the position where the hand needed an unbiased estimation of $196ms$ (on average) to catch up. It indicates that a saccade typically makes an estimation of the future position of the hand in point-to-point reaching tasks.

Continuous tasks such as tracing and tracking typically involve a sequence of saccades, where the leading effect of gaze cannot be simply defined by a single discrete saccade. Delay found by cross-correlation on each component (horizontal, vertical and depth direction) has been widely used as a benchmark method (Mrotek et al. 2006; Gielen et al. 2009; Tramper and Gielen 2011). This method normalises data by subtracting the statistical mean of the data and applies a Hann window before cross-correlation which helps to eliminate constant spatial noise. However, this method yields results in each component due to limits brought in by using cross-correlation. Typical values of gaze lead time found by this method are $23 \pm 38ms$ in azimuth, $42 \pm 28ms$ in elevation, and $266 \pm 175ms$ in depth for a tracking task; $220 \pm 125ms$ in azimuth, $230 \pm 125ms$ in elevation, and $390 \pm 180ms$ in depth for a tracing task (Gielen et al. 2009). Tramper and Gielen (2011) further analysed the total lead time calculated by cross-correlation by modelling it as the sum of saccadic lead time and primary lead time. This work updated the average time that gaze led the hand in tracking tasks of $28 \pm 6ms$ for the frontal plane, and $95 \pm 39ms$ for changes of vergence, and in tracing tasks, lead time of $287 \pm 13ms$ for the frontal plane and $151 \pm 36ms$ for changes in depth. It also demonstrated a constant spatial lead of gaze of about $2.6cm$ in 3D visuomotor transformations, which corresponded to about 2° in visual eccentricity.

Previous research has focused on eye-hand coordination studying user behaviour patterns where human computer interaction was not their primary concern. However, people interact with the computers or virtual worlds through a medium, such as pointing devices or various sensors, which is involved in indirect human-computer interaction rather than direct use of physical hands, although direct input devices (e.g. touch screens) are exceptional on this matter. Thus, the users may behave differently from their natural habits in the interaction tasks. For instance, Wang and MacKenzie (1999) have found increasing orientation time for graphic-to-graphic matches and spatial errors for physical-to-graphic matches when haptic devices were involved.

A computer mouse as a typical indirect input device has been intensively studied in terms of indirect eye-hand coordination, or to be more specific, gaze-mouse coordination. It has been proven that gaze and mouse cursor positions have a strong relationship in discrete actions of human-computer interaction tasks, with gaze leading mouse movements. Chen et al. (2001) have found in web browsing, 84% of the screen regions the mouse lingered were also visited by gaze; further, there was over a 75% chance that a mouse would move to a meaningful region that was very close to the gaze. This suggested that a mouse cursor could be an alternation to show regions of interest in web browsing. Moreover, active mouse movement patterns have been explained in browsing web search pages, which tended to follow the gaze position vertically, as well as marking a particular region of interest (Rodden et al. 2008). A preliminary study has been undertaken for predicting gaze positions by analysing mouse cursor positions based on a combination of findings from previous studies in web searching (Guo and Agichtein 2010).

However, apart from the typical pattern that gaze leads mouse, additional patterns have also been reported. Smith et al. (2000) examined two simple target pointing tasks that required subjects to select two fixed targets alternately, and to select a target presented at random locations. They found there existed several different gaze-mouse coordination patterns that gaze not only led the mouse cursor but also directly followed the cursor, or switched between target and cursor. Bieg et al. (2010) studied three target search and selection tasks where the subjects were asked to find a single target, or a known target from a grid of targets, or an unknown target from a pile of randomly scattered targets. They found that when the target was unknown, the subjects consistently followed their gaze with the mouse cursor, but when the target was known, the eyes fixated on the target rather late. Liebling and Dumais (2014) further inspected gaze-mouse coordination using a variety of target types and applications. They reported that the gaze led the mouse only about two-thirds of the time, and the leading or lagging effect of the gaze depended on the type of target and familiarity with the application. However, the cause remains unclear.

There are few studies discuss how indirect input devices correlate with eye movement in continuous manipulation, which I will elaborate in Chapter 3.

2.3.2 Multimodal integration paradigm

Manual input techniques include mouse, touch screen, mid-air gesture control, and haptics. A computer mouse is a relative indirect input device. Camera-based mid-air gesture control uses absolute indirect input techniques while other mid-air gesture control that applies hand-attached accelerometer/gyro sensors can be considered as relative indirect input techniques. Haptic devices are absolute indirect inputs. Typically, a touch screen is an absolute direct input device, but it can also be used as a relative indirect input device depending on the design of the interaction techniques. Studies have already integrated these manual inputs with the eye tracking technique. The basic idea of the gaze modulated multimodal input follows a two-step paradigm, i.e., firstly using gaze to locate the target or to narrow down the whereabouts of the target, and then using manual inputs to manipulate the target depending on the task. This paradigm naturally solves the Midas Touch problem as a hand confirmation is more explicit than, for example, a dwell eye gaze. This section gives an overview of the current state-of-the-art in these gaze modulated interaction techniques.

Gaze and mouse The fine grained accuracy of mouse pointing is complementary to the coarse gaze pointing. A mouse always has a virtual cursor whose movement is coupled with the gaze to accelerate the cursor movement in gaze modulated interactions, mostly by cursor warping. MAGIC (Zhai et al. 1999) is a typical example that applies gaze selection with mouse pointing. It warps the mouse cursor to the vicinity of where the gaze is and then uses the user's hand to achieve fine selection. One problem with this approach is that the cursor warp is triggered by a mouse movement after some time of inactivity, the cursor tends to overshoot because the mouse is already in motion when the warp is initiated. Zhai et al. (Zhai et al. 1999) suggested a compensating solution based on the initial motion vector and the distance vector, while Drewes and Schmidt (2009) tackled this issue by using a touch-sensitive mouse instead of a normal mouse, so the cursor warp was not triggered by detection of a mouse movement but using a touch on the mouse as a signal to start the warp. Similarly, Rozado (2013) activated the mouse cursor warping by a manual cue such as a keyboard event in order to prevent unexpected cursor warping. This way eliminates the overshoot problem because the mouse stays still when the warp started, and the trigger is clearly sent by users so it can also avoid unexpected mouse movement.

These studies also confirmed the efficiency improvement using MAGIC compared to the traditional manual pointing in various tasks including target acquisition, text selection, text cursor positioning, and drag-and-drop operations.

Gaze and touch Although a touch screen is an absolute direct input device, when it is combined with eye tracking, it can be used as a relative direct input device instead. The reason is that a virtual cursor is usually unnecessary in direct inputs, so using gaze can convert direct inputs to indirect inputs as the gaze, instead of the fingers, indicates the region of interaction, and the gaze position does not overlap with the finger position at most of the time. A touch screen can introduce relatively accurate pointing which is much better than gaze pointing but not as fine as a mouse. The advantage of touch screens is the multi-touch controls which is more natural than the conventional mouse manipulations. Besides, converting to indirect inputs can also mitigate the issue of hand occlusion in direct inputs. More differences between traditional direct touch and gaze modulated indirect touch techniques were summarised by Pfeuffer et al. (2014).

The gaze modulated touch screen technique was initially investigated in remote displays where a touch screen works as an external indirect input device. Stellmach and Dachselt (2012) designed and evaluated the Look & Touch technique for 2D object selection on a remote screen at different sizes and distances. They further designed the Still Looking technique (Stellmach and Dachselt 2013) that extended the gaze-supported selection to manipulation of remote 2D targets. Their design took advantage of the touch screen to facilitate the interaction. For example, the users could slide on the touch screen to adjust the size of the selection mask, or swipe to cycle through the possible selection candidates as the disambiguation strategy. Simeone (2016) compared the performance of direct and indirect touch in stereoscopic displays and the results indicated that indirect touch interaction techniques provided better viewing experience than conventional 3D interaction techniques.

Pfeuffer et al. (2014) discussed four 2D Gaze-touch applications on the touch screen itself instead of a remote screen, including image gallery, paint, map navigation and multiple objects manipulation. These applications applied indirect touch or a combination of indirect and direct touch. For example, indirect-rotate-scale-translate (RST) enabled common multi-touch RST manipulations without direct touch on the images; remote-colour-select enabled colour selection without direct touch on the expected colour but just a looking with a manual tap confirmation from anywhere on the screen. This work also enables seamless transition between direct touch and indirect touch. The Gaze-shifting technique (Pfeuffer et al. 2015) for transitions between direct and indirect inputs using a pen on touch screens was further introduced. A three-point interaction technique that combines bi-manual direct touch and gaze was investigated by Simeone et al. (2016).

Gaze and gesture Mid-air gesture control facilitates a natural interaction technique, especially in 3D interaction in which using a mouse or a touch screen is not as intuitive. This advantage specifically benefits its applications in VR. Moreover, gestures carry rich information so it is even used as an alternative to languages. Although pointing with gesture control is not as accurate as using a mouse, it is still better than gaze pointing. However, mid-air gesture control is prone to arm fatigue, particularly in continuous manipulation as the arm needs to be constantly held in the air. Further, unlike touch gestures, lacking haptic feedbacks may degrade user experience when using mid-air gesture control.

Mid-air gesture control provides an indirect interaction technique. Depending on which type of input devices was used for 3D tracking, it can be either relative or absolute. The relative technique is to attach the tracking device to users' hand such as the Wii Remote controller. It realises 3D tracking using accelerometer or gyroscope sensors. Because the sensor is attached to the users' hand, it is similar to the case of using a mouse. Pouke et al. (2012) combined eye tracker and mid-air gesture interaction using a 6-DOF accelerometer/gyro sensor attached to the users hand to perform gesture control. The eye gaze was used for object selection and no cursor of the hand tracker was mentioned particularly. Because the gesture sensor was attached to the user's hand, the gesture detection area was always centred on the physical hand position. Thus, it does not need absolute mapping between the device and the display space.

However, attaching a device to the hand will increase arm fatigue which is already a problem for mid-air manipulation. To avoid this problem, modern gesture recognition trackers, such as Kinect and Leap Motion, adopt the more unobtrusive camera-based technique. These trackers can be simply placed near the desktop but the camera-based feature requires a spatial mapping which makes them absolute pointing devices.

Yoo et al. (2010) developed a 3D user interface for large-scale displays using head orientation as an alternative to the gaze. The angle of the head indicated attended regions on the display to apply bimanual mid-air gesture commands. This application enables distant gestural control on large-scale display with a quick acquisition. The gesture commands designed in this study was discrete, such as push and pull for zooming in and out. Hales et al. designed a system that used gaze to select object and hand gestures for making discrete commands, such as extending two fingers for toggling the switch of an infrared light. Song et al. (2014) discussed a computer-aided design (CAD) application that used hand gestures for basic manipulation such as translation, zoom and rotation. The application only applied the eye tracker to assist zoom by using the gaze position as the centre of zooming. Slambekova et al. (2012) reported a framework using a "look at" mechanism for choosing objects, namely that hand gestures were used to trigger the selection and de-selection while eye gaze was used to determine the object on which to apply the selection. In their study, the objects could be translated, rotated, and scaled by 3D gestures once selected.

Chatterjee et al. (2015) summarised the gaze+gesture technique in two phases, target acquisition using gaze and target action using gesture control. They discussed three scenarios that have the gaze+gesture technique adopted, desktop, word processor and 3D model viewer. The usability evaluation revealed that this combination outperformed gaze only and gesture only interactions, and reached equivalent performance with mouse and trackpad. Velloso et al. (2015) investigated the 3D selection performance of gaze, 2D cursor with raycasting, and 3D cursor using gesture control. The results show the gaze is faster than the other two. They also found that the selection confirmation time was longer when the selection was followed by manipulation than when it was not.

Zhang et al. (2015) investigated the usability of combining gaze and mid-air gesture in remote target selection on a large display. Their results show positive feedbacks in terms of user preference comparing to gesture-only interactions. However, they also reported that gaze was prone to selection errors due to the fact that the gaze moved faster than the hand so the gaze might move away before the termination of the hand action.

Gaze and haptics Haptic devices provide an additional perception modality of kinematic, force or tactile feedback which benefits users with better cognition of how they performed in a task and helps them improve their performance in an intuitive and natural way. Especially for some tasks that rely largely on haptic feedback, merely visual feedback cannot be as efficient or even causes errors. For instance, endoscopic surgical training is extremely difficult to achieve expected results without haptic feedback (Kincaid and Westerlund 2009). Haptic feedback can also benefit hearing or vision impaired users with a better lifestyle.

The most common tactile feedback is vibration which is widely embedded with the devices in our daily life such as mobile phones, mice and game pads. However, vibrations are typically given in discrete events. Other haptic devices can simulate finer force feedbacks within continuous kinematics, such as a stylus pen, haptic gloves or even aerial haptic in free air(Sodhi et al. 2013).

On mobile phones or wearable devices, vibration is typically given as a confirmation of gaze gesture interactions. Note the gaze gesture is different with the aforementioned gaze+gesture technique. Gaze gesture is only related to gaze movements while gaze+gesture involves both gaze and hand gestures. For example, Kangas et al. (2014a) proposed an interactive method for using gaze gestures as an input method with vibrotactile feedback as confirmation of the gaze event. They designed four gestures using gaze strokes for a contact list browsing task, which were scrolling up and down, selecting, and cancelling. Haptic feedback was given in four different conditions to assess how it would impact on user performances. Those were no haptic feedback, only haptic feedback when stroking from outside the device to inside the device, only haptic feedback when stroke from inside the device to outside the device, and full haptic feedback. The results showed improvement of the gesture performance with fewer errors, especially when gaze stroke moving from inside the device to outside.

Rantala et al. (2014) introduced a pair of gaze gesture eyeglasses with three haptic actuators, one on each end of the glasses frame legs, and one on the bridge. They conducted two user studies to find out the accuracy of distinguishing stimulations from the three actuators, and the timing of haptic feedback the users preferred to use during gaze gestures. The results showed that the accuracy of one actuator outperformed two or more actuators and it was in line with the preference of the users. The haptic feedback was useful mostly at the first stroke of gaze gestures. These glasses could be applied in VR/AR applications that focus more on mobility. Kangas et al. (2017) further studied the impact of haptic feedback on gaze gesture performance. The result showed that gaze gesture was faster and more stable when haptic feedback was given. They also found a marginal effect that longer duration of haptic prompts led to longer duration of gaze gestures.

Post-stroke rehabilitation and virtual surgery involve continuous movements, in which we can apply the two-step paradigm using gaze selection followed by kinematic manipulation. Kinematic and haptic guidance with force feedback can help the patients to practice passively in post-stroke rehabilitation. Frisoli et al. (2012) proposed an attention-driven multimodal architecture for upper limb stroke rehabilitation using eye tracker and robotic exoskeleton. This system consisted of four components: 1) an arm exoskeleton for guiding patient's right arm with force to accomplish reaching tasks; 2) an eye tracker for 2D object selection with gaze; 3) a Kinect for 3D object tracking, selection and communication with the exoskeleton; and 4) a BCI (Brain-Computer Interface) module for estimating a patient's motor intention with motor imagery. The BCI module mainly applied an EEG classifier for discriminating brain activity for right arm movement intention and the rest. Based on the output of the BCI classifier, the eye tracker will select the target object and send it to the Kinect, which calculates the depth and location information for the exoskeleton to make the kinaesthetic movement plan.

Minimally invasive surgery (MIS) is a surgical procedure performed by entering through a small incision with long thin tools to achieve less tissue damage and equal treatment results. The procedure is conducted with the aid of a camera to provide the view of the operation area. This process requires intensive practice for hand motor dexterity and eye-hand coordination. Some surgical simulations are facilitated with master-slave robots in teleoperation or telesurgery. Its effectiveness is often limited by the lack of haptic sensory when operating with a remote robot. In both scenarios, haptic feedback is a crucial element for surgeons' safe performance. Eye tracking provides a way for forbidden-region virtual fixtures (FRVFs) which helps surgeons to locate target tissue with a safety margin to prevent injury to other structures (Rosenberg 1993).

Mylonas et al. (2012) proposed two FRVF methods, Gaze-Contingent Motor Channelling (GCMC) and Gaze-Contingent Haptic Constraints (GCHC). GCMC describes the concept that a dynamic force exerts from the haptic tooltip towards the position of gaze in planar manual tracking as shown in Figure 2.3a. The tracking accuracy has been tested in a task that tracks a target on a mesh with regularly deformable patterns such as heartbeat, where the target moves with the deformation movement. GCHC extends the GCMC framework into 3D manipulations. A binocular eye tracker was integrated into this method that provided the availability of depth information. The haptic constraint reflected in that the exerted force was proportional to the distance between fixation point and tooltip within a small pre-set range. The force maintained constant outside of this range, it formed a tube-like force field for each target on the mesh surface, see Figure 2.3b. A planar hard boundary was also introduced at a small distance from the mesh surface for safety purposes. They developed a shooting game with three stages to test the techniques, the first stage had no constraints or force, the second stage needed aiming purely with the gaze, and the third stage had GCMC fully engaged. The user study shows improved concentration on task learning quality of novices when force feedback was involved. James et al. (2013) further verified the learning advantages of GCMC compared with “free-hand”.

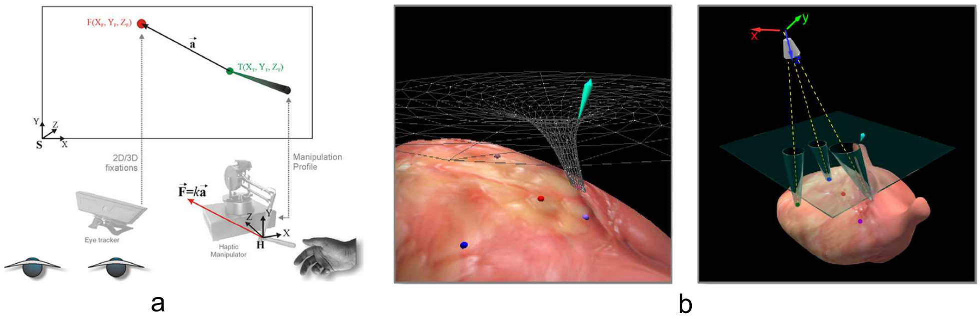


Figure 2.3: *Visual fixture frameworks proposed by Mylonas et al. (2012) in MIS. (a) Illustration of GCMC framework. Eye tracker localizes the 2D/3D fixation F of the user on a screen or stereoscope. Virtual tool T is achieved through a haptic manipulator. Depending on the Cartesian distance between F and T, a force toward the fixation point is exerted on the hand of the user via the haptic manipulator. (b) Illustration of GCHC. The fiducial markers are locked that can only be accessible through the pathways with virtual tool. The hard planar provides a safety boundary.*

2.4 Challenges and Existing solutions

Combining gaze with hand control, on the one hand, helps solve the Midas Touch issue, on the other hand, gaze introduces indirect interaction into hand control which can mitigate hand occlusion and arm fatigue problems. However, gaze modulated pointing inherits a problem of eye tracking, that is fine pointing inaccuracy (proximity problem). Furthermore, the lack of depth accessibility of eye tracking makes it inconvenient when an occlusion occurs (occlusion problem). Both can cause selection ambiguities in 3D interactions.

Fundamentally, gaze pointing is similar to mouse pointing as both interact with the virtual world through a 2D point on the screen, so the 2D point intersecting with the exact target object is essential for selection accuracy. For example, in Figure 2.4 (a) and (b), when two objects are very close to each other, it can be challenging for eye trackers to select the expected target. Here, I define these two cases as the *proximity problem*. In the multimodal context, this problem can be counterbalanced using virtual pointer based hand control, such as a mouse; but in natural interaction using the virtual hand metaphor, such as mid-air gesture control, it can remain as a problem because current hand tracking techniques cannot achieve comparable accuracy to mouse pointing.

As aforementioned, ray-casting is the most common method to extend 2D pointing to 3D. It is also commonly used in gaze pointing. However, when an occlusion occurs, the ray can intersect with several objects. Because eye tracking lacks depth accessibility, it is confusing which object is the expected target. Moreover, occlusion also has a visibility problem, as shown in Figure 2.4 (c), the target may be invisible from the scene. Here, I define this case as the *occlusion problem*.

In this section, I review the existing solutions for both problems, which are referred to as disambiguation mechanisms or disambiguation techniques later in this thesis.

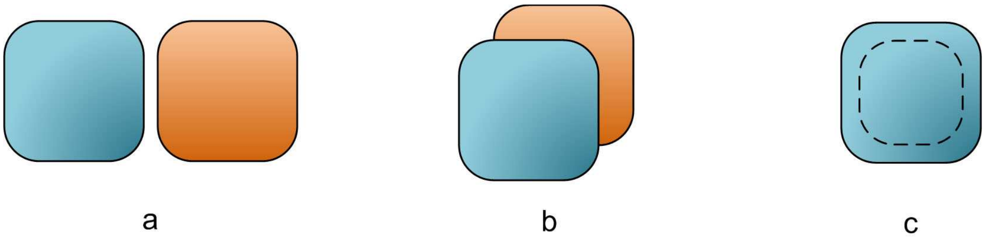


Figure 2.4: *Object interaction. (a) Proximity. (b) Partial oclcusion. (c) Full occlusion. The dashed outline represents an occluded object.*

2.4.1 Proximity problem

Stellmach and Dachselt (2012) categorised the following solutions for the inaccuracy issue of gaze interaction. These methods are specifically developed for selecting small targets from crowded clutters, which is essentially the same issue with the proximity problem, so they can share the solutions.

Manual Fine Selection As I briefly mentioned, the virtual pointer metaphor can help achieve fine pointing accuracy. The workflow is that gaze provides a coarse estimation of the target location while some hand-based manipulation, such as a mouse, realises fine pointing so that combining gaze and hand-based manipulation can compensate the gaze inaccuracy. It follows the idea of a two-step coarse-to-fine selection. In the first step, the gaze conducts a coarse selection, then in the second step, the manual input enforces the fine selection.

The MAGIC pointing technique (Zhai et al. 1999) is based on this idea that firstly warping the mouse cursor to the vicinity of the gaze position, then using the mouse to finish the fine selection. Stellmach and Dachselt (2012) adopted the MAGIC idea into their touch screen controlled distant display to achieve fine grained selection. The gaze-directed cursor firstly determined a rough area of interest, a selection mask centred on the cursor was then enabled. The mask contained all possible targets close to the gaze position. The users could move their fingers on the touch screen anywhere to apply a relative movement of the cursor on the distant large screen to select the expected target inside the mask. The users could alternatively swipe across the touch screen to cycle through all possible targets contained by the mask to find the expected target. The idea of using a mask instead of a point is an example of volumetric probe I will explain in the solutions of the Occlusion problem (Section 2.4.2)

Magnified Target Because of the limited accuracy of the eye trackers, an intuitive solution is to enlarge or magnify the virtual target either visibly or invisibly. The most common technique is also a two-step magnification (Lankford 2000) who follows a coarse-to-fine pattern. It divides the point-and-select task into separated pointing and selecting operations. The first step locates the surrounding area of the gaze and pops up a magnified view of this area. In the second step, if the target is inside this magnified area, the selection can be activated by a precise pointing on the target.

Kumar et al. (2007) implemented the EyePoint technique based on the two-step magnification combining eye gaze and keyboard triggers. The magnified view is triggered by manually pressing a hotkey. Using manual confirmation other than gaze dwells can help eliminate the Midas Touch problem so that unintentional selection activation can be prevented.

Stellmach and Dachselt (2012) proposed two techniques using a zoom lens to present the magnified view when interacting with a touch screen. The first technique activates a zoom lens attached to the gaze movement after a manual tap on the touch screen. The zoom lens is always centred on and constantly followed the gaze position. In the other technique, the zoom lens does not update its position until the gaze goes beyond the boundary of the lens. Otherwise, the zoom lens stays still and the gaze can freely move inside it. Both techniques employ a vertical sliding gesture on the touch screen to adjust the zoom-in factor.

It is worth noticing that only visually magnifying the targets without introducing more object occlusions can help alleviate the inaccuracy of gaze selection. Enlarging the actual size of the target without relocating them might make the problem worse. Miniotas et al. (2004) applied a technique that invisibly expanded the bounding box of the target so that the visual display of the object remained the same but the valid selection area was much larger. This technique improves the efficiency and accuracy of gaze selection with isolated small targets, but in our concerned scenarios, the expansion may cause even more severe intersections of the ambiguous targets, and the problem remains unsolved.

Target Estimation The prediction approach using behavioural data is used to model the user's visual attention. Therefore, the noisy gaze data can be used to estimate and correct the object of interest. For example, Salvucci and

Anderson (2000) described a probabilistic algorithm in order to interpret gaze focus in a WIMP (Window, Icon, Menu, Pointer) example. In this method, the probabilities of producing a gaze at a certain position are calculated based on the intention to attend each item. Then the prior probability of attending each item is calculated based on a given prior score assigned to different WIMP widgets. Combining both the current state and context of the environment can give the best estimation of the object of interest. This method indicated that using the semantic meaning of the visual contents can improve the prediction performance. Alam and Jianu (2016) adapted this work in a structured visualisation application to detect the viewed object, and the results showed that the intelligent interpretation using context information could yield better detection accuracy.

Knowing which point on the scene is actually in attention can help correct the calibration offset caused by the eye trackers (Okoe et al. 2014; Pfeuffer et al. 2013).

As the virtual hand metaphor also suffers from the proximity problem, similar approaches were developed. Periverzov and Ilies (2015) presented the IDS technique to achieve accurate selection of small objects using users' behavioural cues to infer the target object. Specifically, they use the action efficiency cue to estimate the effort, and the action persistence cue to estimate the level of perseverance required to select a particular object. De Haan et al. (2005) developed a scoring function to get the most possible intended object depending on the location of each object with respect to the axis and apex of the selection cone, previous scoring values and other adjustable factors. The smart ray (Grossman and Balakrishnan 2006) technique uses an algorithm to determine the target weights which are obtained based on the objects' proximity to the ray cursor. The closer the centre of an object is to the ray, the larger its weight is.

2.4.2 Occlusion problem

Because the occlusion problem consists of both visualisation and ambiguity issues, the solution to this problem is essentially visualisation techniques. Based on Elmqvist and Tudoreanu's taxonomy of 3D occlusion management (Elmqvist and Tsigas 2008), there are five types of solutions to improve the visibility of the occluded objects: multiple views, transparency, distortion, volumetric probes, and tour planner.

Multiple views An intuitive reaction when a full occlusion occurs is to change the viewing perspective. In 3D modelling software such as CAD and Maya, multiple views in three orthogonal perspectives are provided, as well as an interactive way to manually rotate the model or the camera. Guidelines of multiple views system design were presented by Wang Baldonado et al. (2000).

Transparency The idea takes advantage of transparency to reveal the occluded object. The basic concept is to directly remove part of the occluding layer to show the details inside, especially in complex furniture layout, anatomy and engineering graphs (Li et al. 2007). An interactive way is to allow users to cut holes into the occluding object by themselves (Coffin and Hollerer 2006). In order to retain the geometry information of the cut-away layer as a reference, semi-transparency (Chittaro and Scagnetto 2001) or phantom outlines of the transparent objects (Diepstraten et al. 2002) can be applied.

Distortion Usually, a linear projection is used to display a 3D virtual scene, typically perspective or parallel projection. An occluded object in one projection may be seen in another, so distortion uses this projective difference to reduce occlusion. The simplest way is to change the projection method of the whole scene. For example, Elmqvist and Tsigas (2007) applied an animation to switch from perspective projection to parallel projection when occlusion occurred. The more complex techniques are to compose multiple projections in one scene (Agrawala et al. 2000; Singh and Balakrishnan 2004).

In particular, the fisheye projection can also help solve the proximity problem. In a 2D scene, applying a fisheye projection at the region of interest can distort this area by spherically popping it out, so the object in the focus of the fisheye centre will be magnified for easier selection (Ashmore et al. 2005).

Volumetric probes This method utilises a volume instead of a point to coarsely select a set of candidates among which the final target is included for later fine selection. It conforms to the two-step coarse-to-fine pattern. Similar to the mask (Stellmach and Dachselt 2012) we mentioned earlier, there are several alternatives of the volume, for example, cone (Steed and Parker 2004; Steed 2006), spotlight (Liang and Green 1994), aperture (Forsberg et al. 1996) and sphere (Elmqvist and Tudoreanu 2007; Kopper et al. 2011; Periverzov and Ilies 2015). This mechanism provides better accuracy with a sacrifice of one more step of coarse selection, followed by a rearrangement of the coarsely selected candidates, typically repositioning them to avoid occlusions and proximity. Note that the aperture is a multimodal modification of the spotlight technique which sets the point of origin at users' eye and the conic volume can be adjusted by users' hand. The size of the conic volume is determined by the distance between the eye point and a circular aperture cursor between the eye point and the image plane. A user can move the aperture cursor in or out towards their eye to change the distance and thus the conic volume.

The geometry of the volumetric probe helps define how the cluttered objects should be mapped to their new positions, such as the spherical BalloonProbe which makes the objects scatter spherically and the wedge-shaped Balloonprobe makes the objects relocate in the shape of the wedge (Elmqvist and Tudoreanu 2007). The probe is usually attached to the cursor and the user can move the probe to select where a clutter of objects needs to be rearranged. Users also have the flexibility to control the size of the probe, which can determine how apart is necessary for removing the occlusion ambiguities.

However, the repositioning of the potential targets is not restricted to the shape of the probe. For example, SQUAD (Kopper et al. 2011) repositions the objects in a quad-menu, Ren and O'Neill (2013) repositions the objects in a conic menu, Cashion et al. (2012) repositions the objects in a grid pattern, Wonner et al. (2012) used a flexible starfish shape, and Grossman and Balakrishnan (2006) used a flower shape in the flower ray technique to present the potential targets.

Tour planner Tour planner presents all targets in a scene by precomputing a path through all of them, and then guides users interactively explore the scene following the path. It is not very common in object selection tasks but can be useful in wayfinding and navigation. For more relevant references please refer to the occlusion management survey (Elmqvist and Tsigas 2008).

Among the five categories of occlusion management, distortion and volumetric probes are capable of solving both proximity and occlusion problem. Considering the consistency of interactions, it is desired to use the same interactive pattern under all circumstances, which can preferably reduce user's learning time and confusion during interactions (Mandel 1997). Therefore, it is preferable to use distortion or volumetric probes when both problems may occur. Distortion is preferred to global tasks because it provides more context information while volumetric probe deals with local scope.

Interestingly, gaze modulated hand control follows a two-step paradigm to avoid the Midas Touch problem; in the meantime, the popular coarse-to-fine selection in most volumetric probe techniques also follows a two-step paradigm. These features inspire the design of disambiguation technique in Chapter 5.

2.5 Summary

The spatiotemporal coordination between eye and hand movement provides theoretical support for the development of gaze modulated multimodal interactions. However, there are many different types of hand control devices, such as direct and indirect input devices; and the type of tasks varies, such as discrete and continuous tasks. Therefore, it is important to verify the found eye-hand coordination patterns still applicable to other input devices and tasks. It is particularly interesting to find out if these coordination patterns preserve in indirect input devices within continuous tasks as this thesis is focused on indirect natural hand control such as mid-air gesture controls and haptic inputs.

Moreover, the gaze modulated multimodal interactions using natural hand control inherit advantages from both modalities, but the proximity and occlusion problems are also inherited. It is necessary to properly address these issues to facilitate a more robust interaction technique.

Chapter 3

Gaze Lead

In this chapter, I am going to investigate gaze lead in mouse tracing tasks. Gaze lead is important for designing multimodal interactions because it is fundamental for gaze modulated interactions where gaze is used to inform hand control. Knowing gaze lead can help us design better interactions, for example, to improve interactive efficiency.

Although this thesis focuses on gesture controls, mouse movements were examined in this chapter because mouse and gesture are both indirect input methods. Moreover, mouse is 2D and gesture is 3D. Reducing one dimension helps simplify the experiment and data analysis for an easier start of the research. Current technology also guarantees more accurate measurements of mouse movements than gestures. Besides, NUI emphasises reusing our existing skills, so understanding the innate correlation of eye movements and hand movements in common mouse tasks can shed light on the design of gaze modulated gesture interaction techniques.

As I have discussed in Section 2.3, studies have shown a typical coordination mechanism that eye movements lead physical hand movements in eye-hand coordination. Direct input devices such as touchscreens directly map our hand movements into the virtual environment without any medium, so the eye-hand coordination with direct input devices should be comparable with the eye-hand coordination with physical hand movements. However, indirect input devices map our hand movements into the virtual environment via a medium, and the eye-hand coordination with indirect input devices is interpreted to be the relationship between gaze and the virtual representation of the medium, such as a mouse cursor.

It is proved that this indirect eye-hand coordination still conforms with the gaze lead theory in discrete action (Chen et al. 2001; Rodden et al. 2008; Guo and Agichtein 2010), e.g., point to point transitions where the transition path is insignificant. However, in continuous manipulation where the transition path matters, such as line tracing, there have been few studies that have investigated the coordinated relationship between indirect devices and gaze. In this chapter, I look into this issue using the most common indirect input device, a mouse, within line tracing tasks, to demonstrate the spatio-temporal relationship between gaze and indirect input devices, and to generalise the design implications to gaze and gesture coordination later in this thesis.

3.1 Introduction

It is not always the case that gaze leads hand or virtual cursor. As I previously mentioned in Section 2.3.1, Liebling and Dumais (2014) reported in their study that the mouse cursor led gaze at times where the type of visual stimuli affected the leading or lagging between the two. This uncertainty may degrade the performance of applications that rely on the gaze lead. Imagine, in MAGIC pointing, the cursor warps to the gaze vicinity when the user's hand is leading; the cursor is actually warping away from the region of interest. Therefore, for interactive techniques, designed with the premise of gaze leading, it is critical to distinguish whether gaze leads or the hand leads.

Apart from investigating if gaze lead effects also occur during tracing with a mouse, it is interesting to quantify these effects, both temporally and spatially, because the lead time and position can be used for pre-computation and prediction in tasks with heavy computational load. For example in virtual surgery, rendering a fine meshed object with physical simulation can dramatically reduce the frame rate. Pre-computation and prediction of next frames may help mitigate this problem.

In addition, the differences of lead time and gaze-mouse patterns in previous studies have shown a possible underlying variance due to task complexity. Simple tasks require limited gaze-mouse coordination demands while complex tasks are more demanding. The coordination demands in this thesis are referred to as the perceptual complexity of the trace task. I, therefore, predicted that the degree to which the task required more or less gaze-mouse coordination demands would impact on the variance of gaze lead. Furthermore, the two would be positively correlated, that is, if the complexity of the task increased and gaze-mouse coordination demands increased, the lead time of gaze would increase. Therefore, a task was designed which required participants to use the mouse cursor to pick up a disk and move it along a trace under three conditions of complexity in a 2D virtual environment.

Note that line tracing applications that integrate gaze and indirect inputs already exist, such as the digital pen tracing application designed by Pfeuffer et al. (2015). However, it is developed by discretising the drawing line into segments, so the gaze-integrated manipulations are still point based, i.e., only discrete action is involved.

In summary, the goal of this chapter is twofold; first, to test the hypothesises that gaze-mouse coordination would yield comparable behavioural patterns as eye-hand coordination, in as much as gaze would typically lead mouse movement, and second, that the spatial and temporal lead effect would differ between conditions due to differences in gaze-mouse coordination demands.

3.2 Methods

3.2.1 Participants

Fifteen participants (11 males and 4 females, age 28.5 ± 1.8 (Mean±SE) years) volunteered in the experiments. None of the participants self-reported they had any motor or neurological abnormalities. The participants reported they had either adequate natural visual acuity or corrected vision with glasses. All participants reported to be right-handed and fluent with computer/mouse operations. All the experiments were conducted following the principles of the Declaration of Helsinki and Bournemouth University's research ethics policy, and were approved by the Research Ethics Committee Panel at the Media School, Bournemouth University. Written consent was obtained from each participant after explanation of the experiment.

3.2.2 Apparatus

A desktop mounted eye tracker EyeLink 1000 of SR Research was set up to record gaze movement with sampling rate at $1000Hz$. The spatial resolution was $0.01°$ and the average accuracy was $0.25°-0.5°$. The desktop mount used a chin rest to minimise head movement. Although viewing was binocular, only the right gaze movements were recorded. Participants sat $66cm$ away from the display screen which was a 20" Formac ProNitron 21/750 monitor with a frame rate set to $120Hz$. A mouse with a sampling rate of $120Hz$ and a keyboard was provided for interaction within the experiment. All position data were recorded with respect to the pixel coordinates whose origin was set at the upper left corner of the screen.

Although only the right eye was recorded, it is not uncommon in existing research. A substantial body of research explores eye movement behaviour during a range of visual tasks (Liversedge and Findlay 2000), and the majority of which has recorded the movements from one of the two eyes and typically this is the right eye. In fact Kirkby et al. (2008) reviewed empirical data relating to ocular alignment, during reading and non-reading tasks. They describe how a large proportion of fixations (just over half of the data) are aligned, i.e., fixations are less than a character apart. Clearly, disparity does not occur all of the time and when it does it is very small. Therefore, recording only the right eye does not introduce unnecessary variance comparing to binocular recording.

3.2.3 Stimuli

The tailored task was programmed with C++ and OpenGL. The task scene was presented with a black background containing a small disk with a diameter of $1.56°$, a bordered square box with a side length of $5.64°$ and a border width of $0.1°$, and a predefined trace with a width of $0.17°$. The trace was generated by random hand drawing, and there were three conditions of different complexity levels that required low to high gaze-mouse coordination demands (see Figure 3.1(a-c)):

- Low gaze-mouse coordination demands (LD): a simple straight line, with a length of $24.70°$, no curves, and direction of $33.75°$ upwards from the horizontal line.
- Moderate gaze-mouse coordination demands (MD): a simple curve, with a length of $42.70°$ and two curves.
- High gaze-mouse coordination demands (HD): a complex curve, with a length of $94.14°$ and fourteen curves.

The displays within each trial of the same condition were identical to those shown in Figure 3.1(a-c). The box was fixed at the upper left screen as the destination of the trace. The trace started from the lower right screen and ended at the box. Participants needed to use a mouse to move the disk from the starting end of the trace, and traverse it along the trace until it got into the box.

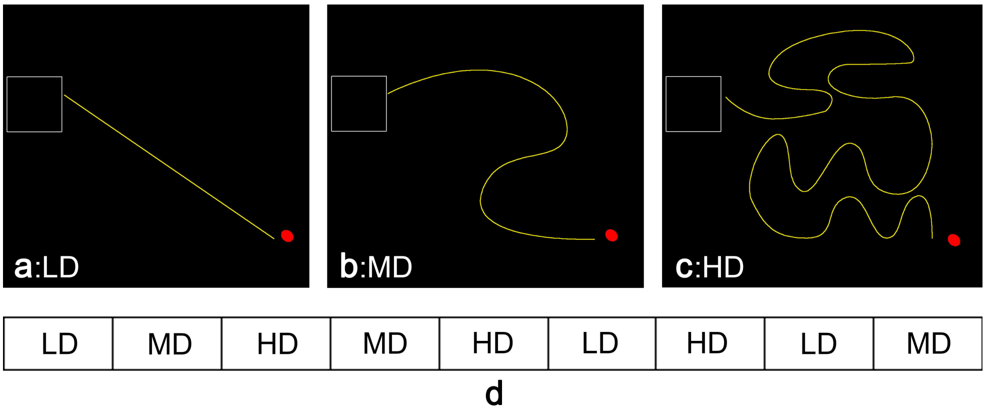


Figure 3.1: *Stimuli for the three conditions of different complexity levels that require low to high gaze-mouse coordination demands. (a) LD the straight line, (b) MD the simple curve, and (c) HD the complex curve. (d) is the order of trial conditions in each trial set.*

3.2.4 Procedure

There were nine trials in one trial set. Each condition appeared once in every group of three. The order of the trials was balanced to prevent the chance that the same trace showed up continuously. All data were recorded in the same order for every trial set as demonstrated in Figure 3.1(d). Note that I did not use randomisation to balance the order because a randomisation among the nine trials may cause the same conditions showing up in a row; a randomisation within each group of three may result in the situation that the last condition of the previous group is the same as the first condition of the next group.

Before each trial started, the screen was blank of background colour with only a little white cross in the middle. To eliminate bias introduced by the initial state, all participants were asked to fix their gaze on the cross before moving onto each trial display. At the beginning of a trial, the disk appeared at a random position on the screen. When the disk was picked up (mouse down and hold), its colour changed to highlight the picked-up status and maintained the highlight colour during mouse dragging. A trial ended when the participant released the disk (mouse up) in the target box after dragging the disk along the trace. Before data recording, each participant carried out one pre-trial of each condition to familiarise themselves with the experiment procedure. Ten trial sets were recorded for each participant. During recording, if the participant veered off the trace, this trial would be discarded and the participant needed to redo it.

Calibration Right eye calibrations were performed binocularly (e.g. during calibration participants viewed the stimuli with both the right and left eyes). The horizontal calibration range was $29.45°$, vertical calibration range was $21.05°$. During calibration, the participant was instructed to stare at one of the nine point grid pattern fixation points. In this process, the initial fixation position was accepted by the experimenter when the pupil appeared stable; the remaining fixation positions were automatically recorded by the calibration system when a stable fixation was detected. The validation procedure was essentially identical to the initial calibration, and on the basis of the initial calibration and validation, the discrepancy between these two data sets was computed. The calibration fixation points extended $0.6°$, and a mean error of $<0.7°$ was accepted as an accurate calibration, and recalibration was performed if the validation error was $>0.7°$. These calibration and validation procedures are standard. The experimental stimuli were presented when a successful calibration was completed. Following nine trials during the experiment, the calibration accuracy was verified, and at that point, recalibration was carried out if necessary. The mean ± SD of validation errors for all trial sets was 0.47 ± 0.14 degrees.

3.2.5 Data pre-processing

There were 1350 trials of 150 sets recorded in total (15 participants, each did 10 trial sets and nine trials in each trial set). By generally reading the trajectories visualised from the data, 9 trials were eliminated from the data processing because of calibration and recording issues, which means 99.3% of the trials were successfully completed. Due to order effect that might be introduced into the data, the first three trials of each trial set were removed from the analysis.

All data were stored on a hard disk for offline analysis with MATLAB (MathWorks). Each trial generated a data file and a screen recording video of the experiment process. The file recorded gaze positions on the screen at $1kHz$, along with start and end time of gaze movement events including fixations, saccades, and blinks. Mouse cursor positions and relative events were also recorded at $120Hz$ with timestamps. For each trial, the start time was defined as when the target disk was moved to the start position of the trace, and the end time was defined as when the disk was moved into the destination box. Mouse movement data were interpolated linearly to match with the frequency of gaze movement data for computational convenience. A Savitzky-Golay filter (Savitzky and Golay 1964) (span = 2% of the total number of data points per trial, degree = 2) was applied for both gaze data and mouse data to remove drifts introduced by blinks and trembles.

3.2.6 Analysis

Distance between gaze and mouse cursor After pre-processing, a data matrix was generated for each trial. Each row represented one sample containing the following parameters: timestamp (t), gaze position x (gpx), gaze position y (gpy), mouse cursor position x (mpx), and mouse cursor position y (mpy). The Euclidian distance between gaze and mouse cursor positions on the screen (DGM) for the i^{th} sample row of a certain time t could be calculated by Equation 3.1:

$$DGM_i = \sqrt{(gpx_i - mpx_i)^2 + (gpy_i - mpy_i)^2} \tag{3.1}$$

The DGM was calculated in pixels then converted to degrees of visual angle.

Lead time of gaze relative to mouse cursor The projection of a gaze position on instantaneous mouse cursor moving direction for a certain gaze-mouse position pair is shown in Figure 3.2. The instantaneous lead time relative to the mouse movement could be defined by Equation 3.2 where the velocity of the mouse cursor (v_m) was obtained by the central differencing scheme as shown in Equation 3.3. Δt is the sampling interval, typically $1ms$. During a fixation, the mouse cursor position would catch up with the gaze position, i.e., the displacement between gaze and mouse cursor (d_{gm}) shortens. Note that DGM is the norm of d_{gm}. Suppose v_m is uniform, t_{lead} will become smaller when d_{gm} shortens according to Equation 3.2. In this case, at the time when a fixation started, t_{lead} had its maximum effect. Hence, only samples at the start point of fixations were examined for providing maximum leading availability.

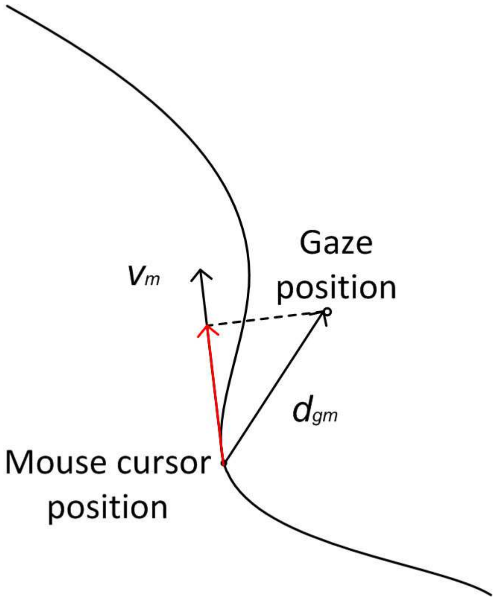


Figure 3.2: *Illustration of all metrics needed for lead time calculation.*

$$t_{lead} = \frac{\vec{d_{gm}} \cdot \vec{v_m}}{||\vec{v_m}||^2} \tag{3.2}$$

$$\vec{v_m} = (\frac{x(t+\Delta t) - x(t-\Delta t)}{2\Delta t}, \frac{y(t+\Delta t) - y(t-\Delta t)}{2\Delta t}) \tag{3.3}$$

Linear regression To test dependency of gaze position and mouse cursor position, linear regression was applied on the x- and y-component of the gaze-mouse position pairs.

The coefficients of the fitted lines have been obtained for each qualified trial, as shown in Equation 3.4 and Equation 3.5, where gpx is gaze position x, gpy is gaze position y, mpx is mouse cursor position x, and mpy is mouse cursor position y.

$$gpx = k_x \cdot mpx + b_x \tag{3.4}$$

$$gpy = k_y \cdot mpy + b_y \tag{3.5}$$

In both equations, k is the slope of the fitted line and b is the intercept. Physically, k indicates the overall rate of change between gaze trajectory and mouse cursor trajectory. We define k as the tracing gain, which is the ratio of the length of gaze trajectory to mouse cursor trajectory. The other parameter b shows the average leading or delay distance.

When k>1, gaze position changes faster than mouse cursor movement; it tends to overshoot while mouse cursor traverses the same amount of distance, and the overall length of gaze's trajectory is longer than mouse cursor's trajectory; when k<1, gaze position changes slower than mouse cursor movement; it tends not to cover as much distance as the mouse cursor covers, and the overall length of gaze's trajectory is shorter than mouse cursor's trajectory; when k=1, the overall length of the trajectories of gaze and mouse cursor should be the same. When b>0, gaze leads mouse movement; when b<0, mouse leads gaze; when b=0, the movements of gaze and mouse are synchronised. A typical case is that k=1 and b>0, where gaze position and mouse cursor position can be superimposed with gaze leading.

Prior to applying linear regression, the data were normalised; this was due to the fact that the lead/delay relationship was highly related to the trace traversing direction. Here we consider the mouse cursor moving direction always conforms to the trace traversing direction. For example, in Figure 3.3(a), when the x-component of the trace/mouse movement v_x goes in the positive direction of the x-axis, if gaze is leading, the coordinate of the gaze will be larger than the coordinate of the mouse cursor ($gpx > mpx$). However, when the x-component of the trace/mouse movement v_x goes in the negative direction of the x-axis (as shown in Figure 3.3(b)), the coordinate of the gaze will be smaller than the coordinate of the mouse cursor if gaze is leading ($gpx < mpx$). If data of the first scenario as in Figure 3.3(a) are fed into the linear fitting, parameter b will remain positive; data of the second scenario as in Figure 3.3(b) will cause b to decrease instead. In this case, the lead/delay effect will be neutralised, leading to the failure of showing reliable results. It is similar to the y-component with respect to the y-axis.

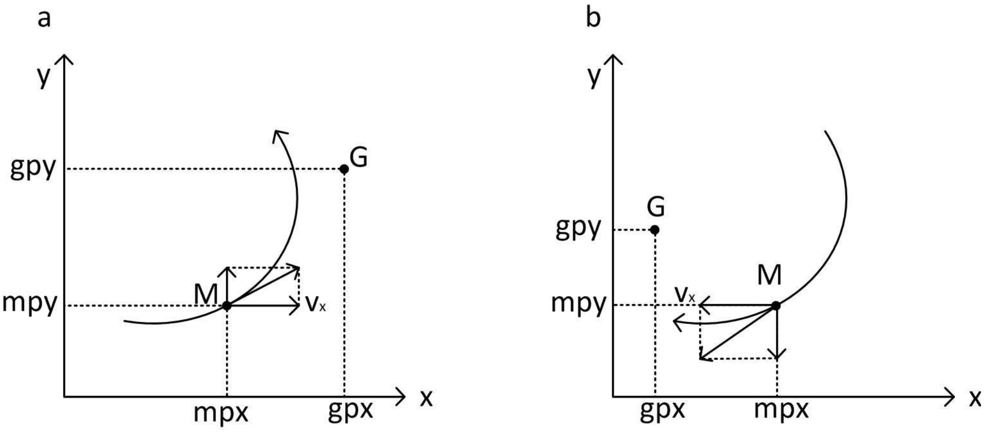


Figure 3.3: *An example of why changing the trace traversing direction impacts on the lead/delay relationship. (a) The gaze G is leading the mouse M movement and the x-component of the mouse movement conforms to the positive direction of the x-axis. (b) The gaze G is leading the mouse M movement and the x-component of the mouse movement conforms to the negative direction of the x-axis.*

Therefore, to normalise the data, the axis directions was used as the reference; if the x-component (or y-component) of the current tracing/mouse direction conformed to the direction of the x-axis (or y-axis), the signs of both the gaze and mouse cursor x-coordinates (or y-coordinates) remained the same on the x-component (or y-component); otherwise, both signs were turned to the opposite.

Statistics To test our initial hypothesis, which was whether there was a significant linear relationship between gaze and mouse movements, a t-test (Paulson 2007) on the regression slope was applied on all sets of coefficients fitted. To test our second hypothesis of whether the level of gaze-mouse coordination demands impacted differentially on any relationship between gaze and mouse movements, a one-way ANOVA was applied for each coefficient, k_x, b_x, k_y, and b_y, yielded from the linear regression, lead time t_{lead}, and gaze-mouse distance DGM. *Post hoc* Bonferroni tests were used to establish differences among the three conditions with different gaze-mouse coordination demands.

3.3 Results and Discussion

3.3.1 Overall gaze-mouse behaviour in tracing

Two types of gaze-mouse behaviour were observed in the experiments. They are visually grouped by the experimenter.

The overall performance of most participants reflected the typical staircase-like gaze movement pattern (Type I) where the gaze position changed rapidly (saccades) then waited (fixations), forming an obvious staircase-like pattern in the plot of x or y coordinates with respect to time. Figure 3.4 shows a typical sample trial of Type I by presenting gaze movement (green solid line), fixation (blue circle), and mouse movement (red dashed line) for each condition: low level of gaze-mouse coordination demand (top row), moderate level of gaze-mouse coordination demand (middle row), and high level of gaze-mouse coordination demand (bottom row) in the frontal plane (left column), x-component (middle column), and y-component (right column) relative to time, respectively.

Another type (Type II) of a smoother gaze movement pattern was observed to occasionally happen in two participants' performance (see Figure 3.5). The time series of gaze movement (middle and right column) did not form the typical staircase-like pattern but continuously followed the stimulus trace more smoothly, where saccades and fixations were not significantly different in the plot. The fixations marked by blue circles are determined by the eye tracker software, where approximately equal counts of saccades were detected for both types in the same condition.

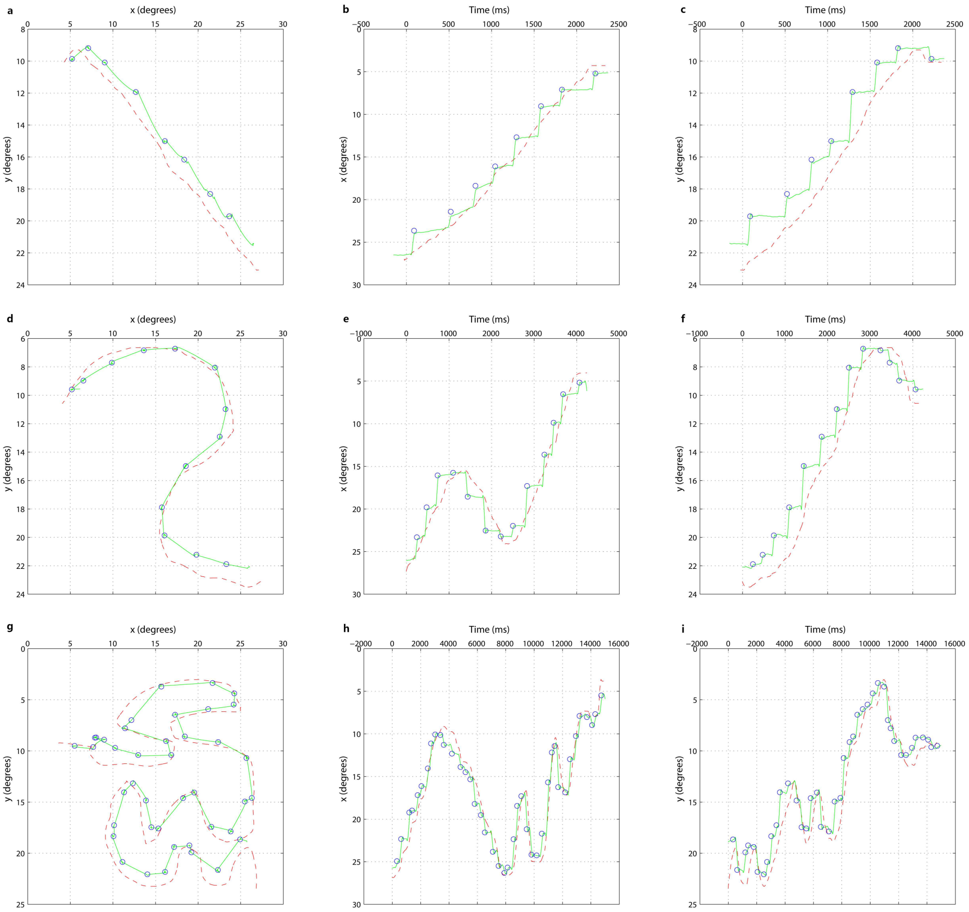


Figure 3.4: *Type I gaze (green solid line) and mouse cursor (red dashed line) trajectories during tracing in low level of gaze-mouse coordination demand (top row), moderate level of gaze-mouse coordination demand (middle row), and high level of gaze-mouse coordination demand (bottom row). The left column shows the gaze trajectory and the mouse cursor trajectory in the frontal plane, the middle column and the right column show the corresponding time series for the x- and y-directions, respectively. Blue circles show the mean position where fixation occurred.*

Comparing the time axes on Figure 3.4 and Figure 3.5, Type II is generally slower than Type I. It appears that the two types of coordination patterns may directly relate to the participants' speed. A possible explanation is that Type II gaze movement was a combination of smooth pursuit and saccadic movement. Smooth pursuit usually occurs when the eyes closely follow a moving object (de Xivry and Lefevre 2007). Research has demonstrated that except for trained participants (Purves et al. 2001), humans are not capable of smooth pursuit without a visible moving target. In this study, the moving

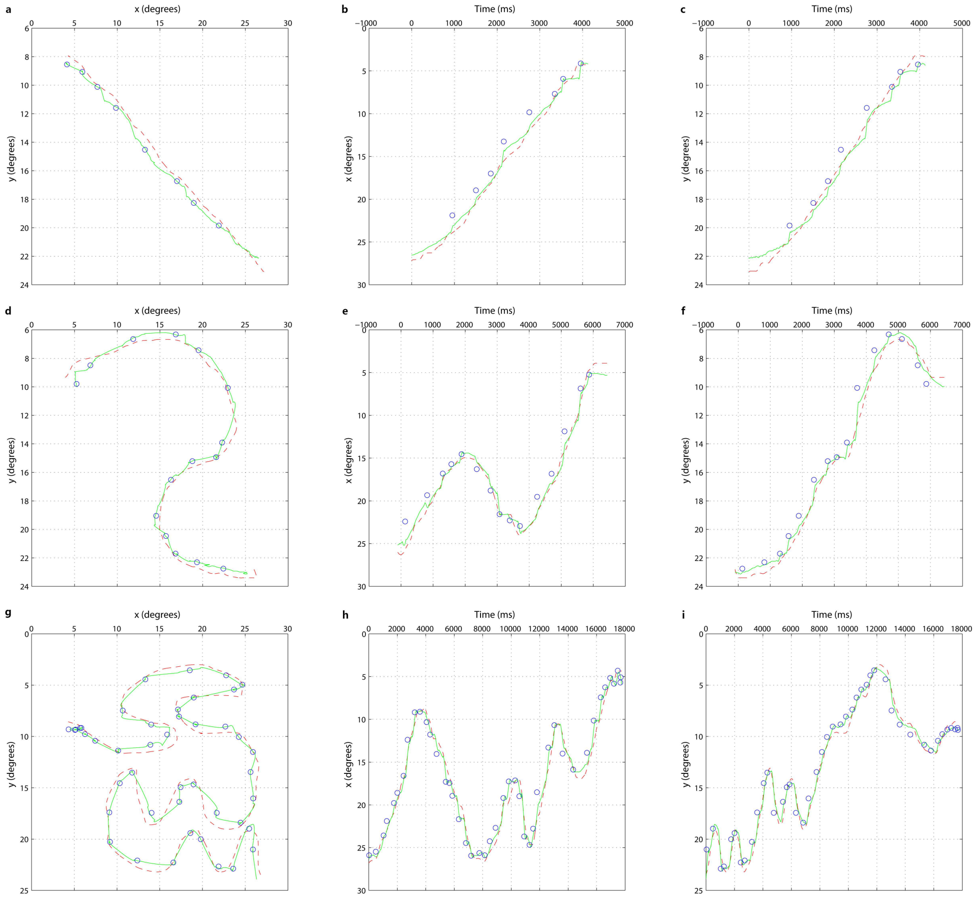


Figure 3.5: *Type II gaze (green solid line) and mouse cursor (red dashed line) trajectories during tracing in low level of gaze-mouse coordination demand (top row), moderate level of gaze-mouse coordination demand (middle row), and high level of gaze-mouse coordination demand (bottom row). The left column shows the gaze trajectory and the mouse cursor trajectory in the frontal plane, the middle column and the right column show the corresponding time series for the x- and y-directions, respectively. Blue circles show the mean position where fixation occurred.*

object was the mouse-manipulated disk. Knowing that the participants were not specially trained for that purpose, it suggested that the participants' eyes were more closely following the movement of the disk. However, when the Type II gaze movements were observed, the gaze was not directly following the movement of the disk but leading it at an extremely limited distance. Therefore, the Type II movements could be due to the predictable movements of the target, the visible trace, and the self-manipulated hand movements. In fact, atypical patterns of saccades were also identified in the data. Specifically, when a fixation occurs following a saccade, instead of staying relatively still, the gaze keeps moving forward with a velocity that is slower and smoother than the saccade. This may explain why in Type II gaze still leads but by a very limited amount. However, it was not the dominant pattern of gaze behaviour observed in the data; so further discussion of the Type II gaze movement is beyond the scope of this study.

3.3.2 Correlation in gaze-mouse coordination

Linear dependency It is shown in the results that gaze and mouse movements are highly correlated in terms of position. To avoid the influence of the trace shape, linear fitting of the coordinates with normalisation which is described in the section *Linear regression* was applied to provide a statistical mean of the gaze-mouse coupling of the x-component and the y-component, respectively.

The tests on the regression slope using t-statistics for the x- and y-components both indicated that there was a significant linear relationship between gaze and mouse movements at the significance level of 5%. The degree of freedom (dfe) of the t-test ranges between [74, 2524] for both x- and y-component. If a t-score of a dfe in this range is greater than 4 according to the t-distribution, the p-value is 0, indicating linear relationship exists. In this analysis, the minimum t-score is 13.79 for the x-component, and 21.82 for the y-component, i.e., t-scores in all trials of both components are greater than 4, so their p-values are all 0, indicating that all tested trials are linear related for both x- and y-components.

Table 3.1 shows the mean and standard deviation (SD) of the coefficients of linear regression that were either generated from all trials or from each condition. Rows ${R_x}^2$ and ${R_y}^2$ show the goodness of fit using the residual variance from the fitted coefficients. An R^2 with a value close to 1 indicates a good fit.

The overall k_x shows that in the horizontal direction gaze traverses about the same distance that mouse cursor traverses. However, in the vertical direction, gaze traverses about 10% less than the distance that the mouse cur-

Conditions	**LD**	**MD**	**HD**	**Overall**
$\boldsymbol{k_x}$	0.99 (0.10)	1.05 (0.07)	1.05 (0.07)	1.03 (0.09)
$\boldsymbol{b_x}$(deg.)	-0.86 (1.34)	0.42 (0.59)	0.60 (0.38)	0.06 (1.09)
$\boldsymbol{R_x}^{\mathbf{2}}$	0.97 (0.04)	0.99 (0.01)	0.99 (<0.01)	0.98 (0.02)
$\boldsymbol{k_y}$	0.90 (0.14)	0.88 (0.13)	0.90 (0.12)	0.89 (0.13)
$\boldsymbol{b_y}$(deg.)	0.14 (0.47)	0.19 (0.36)	0.33 (0.34)	0.22 (0.40)
$\boldsymbol{R_y}^{\mathbf{2}}$	0.98 (0.03)	0.99 (0.01)	0.99 (0.01)	0.99 (0.02)

Table 3.1: *Mean (SD) results of linear regression coefficients for each condition and for all trials.*

sor does. The difference between k_x and k_y suggests that the vertical tracing gain is smaller than the horizontal tracing gain. The directional asymmetry of gaze movement has previously been reported in smooth pursuits where horizontal movements were more accurate and faster than vertical movements (Rottach et al. 1996). Our pattern of results suggests the possibility that such asymmetry also exists in saccadic eye movement.

The low gaze-mouse coordination demands condition shows significance on k_x (both *post hoc* Bonferroni tests with MD and HD showed $p < .001$). The results indicate that in the horizontal direction, the gaze covers slightly less distance than the mouse cursor when tracing with low gaze-mouse coordination demands; but when tracing with moderate to high gaze-mouse coordination demands, the gaze tends to cover slightly more distance. It is not significantly different on k_y among the three conditions ($F(2, 891) = 0.65$, $p = .52$), which indicates that the level of gaze-mouse coordination demands does not affect the tracing gain (k) in the vertical direction.

The results of b_x and b_y show that the gaze lead in both directions grows when gaze-mouse coordination demands increase ($bx : F(2, 891) = 248.87$, $p < .001$; $by : F(2, 891) = 18.67, p < .001$). Surprisingly, the b_x value under the low gaze-mouse coordination demands condition is negative, indicating that the mouse leads the gaze. This suggests that when the task demands less gaze-mouse coordination, the mouse cursor can follow the gaze very closely, which is reflected in the small gaze lead in the vertical direction; or even lead gaze instead, as shown in the horizontal direction. This pattern of results is in line with and extends previous studies (Bieg et al. 2010; Liebling and Dumais

2014). Due to the simplicity of the straight line that requires low gaze-mouse coordination demands, it may be possible that hand motion can be planned at the same time as the trace is initially seen, so proprioception can play the main role in guiding movements of the hand with visual feedback only acting as an accuracy validator. When vision provides the main guiding information for hand movements, i.e. under moderate and high gaze-mouse coordination demands, it mostly relies on an eye-brain-hand interaction. Under low gaze-mouse coordination demands, it seems to be a reversely hand-brain-eye interaction. This interplay can be essential in straightforward tasks or trained tasks when memory, other than real-time vision, stimulates the hand movement.

During tasks that required moderate to high gaze-mouse coordination demands, b_x is larger than b_y. It is assumed that the difference is caused by the shape of the trace. A supplement test was conducted to testify this assumption. The stimuli traces were turned 90° so that the horizontal and vertical movements were swapped. This supplement test data reflected that swap, in which b_y became greater than b_x, but k_x and k_y still remained the same. Therefore, this indicates that b is related to trace shape but not k. In this case, we cannot conclude a general leading distance (b) because the trace shape affects its x and y components, but it gives a hint of the possible shape coefficient defined by the proportions of horizontal and vertical traces, for determining the leading distance on each directional component x and y. According to the fact that all three k values for each condition are similar on each directional component, the tracing gain (k) is not significantly impacted by the trace shape, indicating that other untested trace shapes will share the same result in the same experimental setting.

The standard deviation of the low gaze-mouse coordination demands condition is larger than the other two for all four coefficients. This was caused by the too small number of saccades in some trials of this condition, which was unable to provide enough training data for the linear regression and then led to difficulties in obtaining more robust results.

Correlation between gaze speed and gaze-mouse distance Figure 3.6(a) shows a typical case of the corresponding relationship between the gaze speed (green solid line), the speed of the mouse movement (red dashed line), and the gaze-mouse distance (blue dash-dot line) during a moderate gaze-mouse coordination demanding trial. The peaks of the green solid line represent the saccades; in between the saccades are fixations. Each saccade has one shorter peak on both sides; in other words, at the beginning and at the end of a saccade, there is a small acceleration of the gaze movement (see Figure 3.6(b)). Figure 3.6(c) gives a close-up of a single saccade. The backshoot at the right-hand side of the saccade, which is marked with a red circle, and the dynamic overshoot at the left-hand side of the saccade, which is also marked with a red circle, are clearly seen. The backshoot can be reflected by the first peak of speed in Figure 3.6(b), and the dynamic overshoot can be reflected by the third peak of speed in Figure 3.6(b). The overshoot could be evidently identified in the moderate and high gaze-mouse coordination demands conditions of every participant's performance except for some trials of the low gaze-mouse coordination demands condition and those who have Type II gaze movements. The dynamic overshoot is usually a correction since the target of the initial saccade is not perfect (Kapoula and Robinson 1986). Most studies discussed dynamic overshoots but backshoots prior to the initiation of a saccade are rarely mentioned in the previous literature. Deubel and Bridgeman (1995) reported much smaller backshoots than overshoots but in our specific tracing task, the backshoots are as evident as or only slightly smaller than the dynamic overshoots.

The number of peaks in the blue dashed-dotted line is consistent with the number of saccades. It shows the positive correlation between the gaze-mouse distance and the speed of gaze movement in mouse tracing. It is apparent that the gaze-mouse distance reaches its local maximum at the end of one saccade and its local minimum at the beginning of the saccade, suggesting that the mouse cursor is generally catching up with the gaze position.

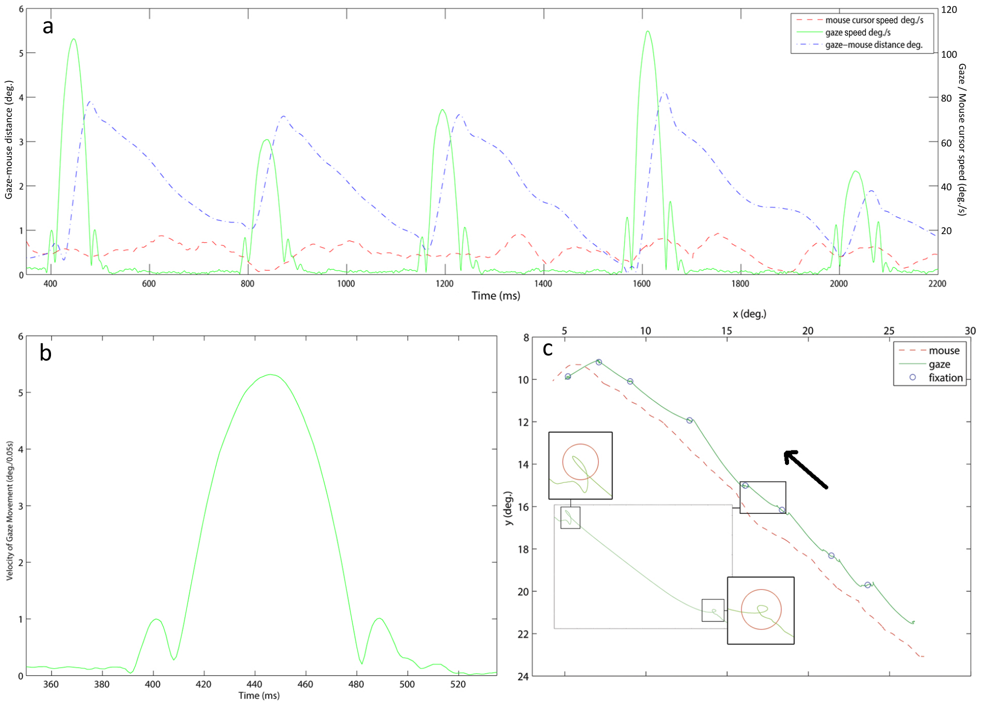


Figure 3.6: *A sample of the relationship between gaze speed, mouse cursor speed, and gaze-mouse distance.*

3.3.3 Lead time in gaze-mouse coordination

The gaze lead effect was validated in the discussion of *Linear dependency*. In this section, a direct evaluation of the lead time for each condition is provided. Table 3.2 gives the lead time obtained by the method explained in Equation 3.2 for each condition. The positive values represent the gaze leading the mouse cursor. The significance test shows smaller average lead time on the low gaze-mouse coordination demands condition (both *post hoc* Bonferroni tests with MD and HD showed $p < .001$). It can be explained by two factors. One is that the mouse velocity of straight line tracing is greater than that of curve tracing. Another is that in some straight line tracing cases the mouse cursor was leading in the horizontal direction but lagging in the vertical direction. This method integrates separate leading/lagging effects in the horizontal and vertical directions to one instantaneous mouse moving direction; so the results are neutralised by the lagging in the horizontal direction. Both factors can be testified in the current data. However, it is unclear to what extent each factor has impacted on the pattern of results. The lead time for moderate to high gaze-mouse coordination demands conditions is in agreement with the

Conditions	**LD**	**MD**	**HD**
Lead time(ms)	223 ± 154	283 ± 113	295 ± 87
$\boldsymbol{DGM}$(deg.)	2.95 ± 2.08	2.95 ± 1.95	2.97 ± 1.91

Table 3.2: *Mean±SD results of lead time calculated by Equation 3.2 and DGM for each condition.*

work of Gielen et al. (2009), indicating that the response time in the central nervous system during finger tracing is similar to mouse tracing.

Figure 3.7 provides an overview of lead time of all participants across all conditions. It is noticeable that some participants have very small lead time in the low gaze-mouse coordination demands condition. The minimum values represented by the bottom of the whiskers are even minus for these participants. It is in accordance with the negative b_x that has been explained in the *Linear dependency* section. Because the lead time is only calculated based on the samples at the end of each saccade where the distance is supposed to be the maximum during each saccade and atypical saccadic movements still occur in Type II, the result of Type II still shows typical lead time.

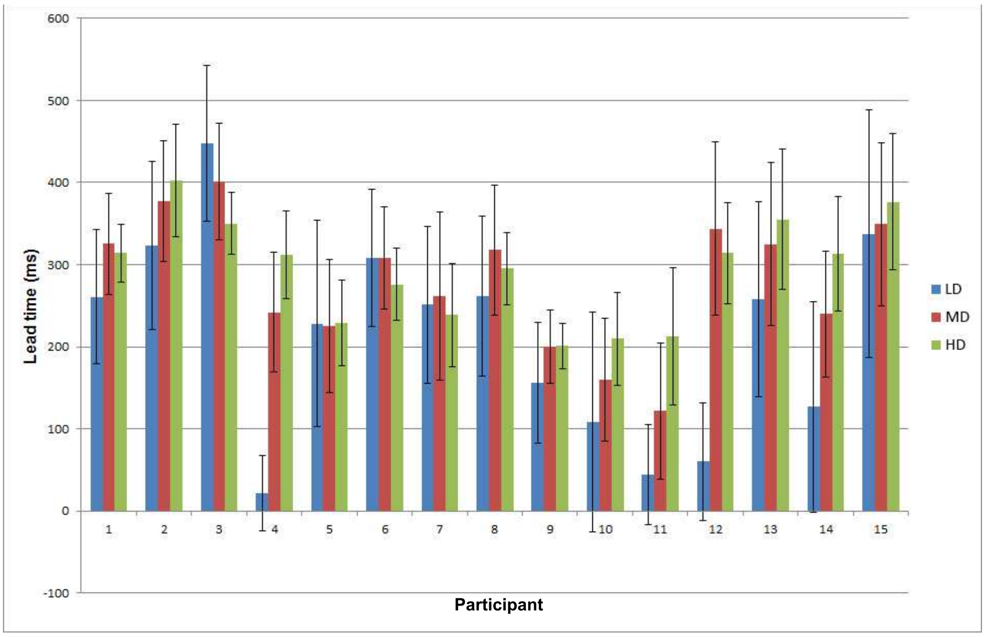


Figure 3.7: *Average lead time for each participant grouped by different conditions.*

3.3.4 Distance between the gaze and the mouse Cursor during tracing

Figure 3.8 shows the distribution of the sampled distance of all gaze and mouse cursor position pairs. The mean ± SD of DGM is 2.96 ± 1.94 degrees for all sampled distances between gaze and mouse cursor position pairs, containing 72.04% of the samples. The mean ($2.77°$) is slightly larger but close to the lead distance of gaze found in the work of Tramper and Gielen (2011), which was $\sim 2°$. The percentage of DGM that falls inside the $10°$ range is 99.36%, which agrees with the findings presented in studies of Binsted et al. (2001) and Bowman et al. (2009). This reflected the observation that the mouse cursor was coupling with gaze by keeping a distance within the high visual acuity area of the retina (which will be explained in the next paragraph). It is worth noting that Type II data showed very small mean DGM that was $\sim 1°$ because smooth pursuit eye movements keep gaze-mouse distance to the minimum while Type I mainly distributed between [2, 3] degrees. The fit of samples only at the beginning of each fixation yields the mean ± SD of DGM to be 3.49 ± 2.22 degrees. Gaze-mouse distance for each condition is given in Table 3.2 which shows no significant difference for the three levels of gaze-mouse coordination demands ($F(2, 891) = 0.76$, $p = .47$).

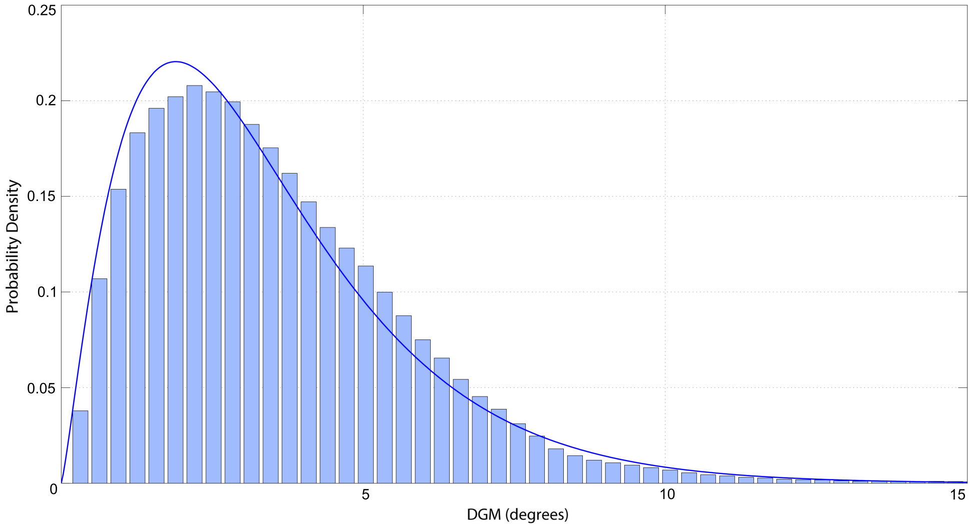


Figure 3.8: *Distribution of DGM of data from all participants and all conditions.*

It is noticeable that the high spatial correlation between gaze trajectory and mouse trajectory does not perfectly guarantee superimposition of the two. For specific tasks like pointing to a specific position, a perfect overlap is possible (Neggers and Bekkering 2000). Previous work built the gaze lead model with the assumption that gaze and hand movements were perfectly superimposed (Tramper and Gielen 2011). However, in tasks like tracing, because it is a dynamic process, it is easier to introduce noise that affects the superimposition of gaze trajectory and trace stimuli, such as complexity of the trace, motor instability, attention shifts, distraction, the size of the motor tool, calibration noise, and personal habits amongst other factors. Our results suggest that the difference is related to the range of visual acuity area. It is well known that human vision achieves the highest acuity at the fovea which extends about 2° around the centre of the retina (see Figure 2.1). The region that circumscribes the fovea for 4° is parafovea. Combining the fovea and parafovea forms an approximately 10° wide area of the retina which we call the visual acuity area. During a fixation, this area takes in the majority of visual information; furthermore, very limited amounts of visual information are taken in during a saccade as saccadic suppression limits the amount of moving information we are able to perceive, so that retinal smearing is not experienced (Irwin 1993). Therefore, fixations and saccades constantly occur in order that we build up a mental representation of the visual environment. The range of visual acuity is wide enough to tolerate the distance between the gaze and the mouse cursor, but the distance beyond this threshold is perhaps not tolerable during this task. The k_y value is about 0.89 as presented in Table 3.1, indicating that gaze trajectory covers only 89% of cursor trajectory, which supports this assumption. Moreover, given that human peripheral vision is good at detecting dynamic movements (McKee and Nakayama 1984), and the only requirement of the task is to keep the disk moving along the trace where too much detailed information is dispensable for completing the task, tracing accuracy in the task was not affected without looking directly at where the disk and the mouse cursor were.

3.3.5 Limitations

This experiment investigated the relationship between perceptual complexity and gaze-mouse coordination but various other factors could be taken into account, such as content saliency. While this is an initial investigation, strong conclusions cannot be formulated, so further investigations into other variables which may also influence the results need to be considered in the future.

Arguments exist that gaze cannot always overtly represent attention or region of interest because of covert attentional orienting (Posner 1980). However, according to perceptual load theory (Lavie et al. 2004), distractor interference can be reduced or excluded from perception when the level of cognitive load in processing task-relevant stimuli is sufficiently high to exhaust perceptual capacity, so gaze is more intensively linked to current attention in a task with a high cognitive load.

Although the order of trials in one trial set has been carefully designed so that the same conditions were not shown continuously, the same order of trials was still repeated in each trial set. As participants repeated the trials ten times, there were practice effects where their performance on the task gradually improved; this was especially noticed in the first condition (the straight line with low gaze-mouse coordination demands). It is also found that their performance fluctuated because of the discontinuity between each trial set. However, by removing the first three trials of each trial set the practice effects and unstable performance became insignificant. A balanced Latin square design (Campbell and Geller 1980) is suggested in this type of experiment to achieve randomisation without repetition.

Occasionally the device lost tracking of eyes such as blinks. These data have been removed from the analysis, but there is a possibility that trivial bias has been introduced into the results due to the incomplete trial data and trace shape sensitivity in lead effect evaluation.

A very large gaze-mouse distance has been observed in several participants' experimental data, possibly because the visual trace and the disk were easy to detect within peripheral vision. According to Fitts' Law (Fitts 1954) and its extension to trajectory-based tasks (Accot and Zhai 1997), shrinking the size of the disk may increase the difficulty of the task and gaze-mouse coordination demands, and correspondingly change the distance between gaze and mouse. Moreover, Jiang et al. (2015) studied the correlation between pupil dilation and continuous pointing task difficulty using Fitts' law to quantitatively define the complexity of the motor task, which can be adopted into this experiment to model the conditions in a more controlled way.

3.4 Summary

In this chapter, I first investigated the spatiotemporal relationship between gaze and mouse movements in a tracing task. To summarise the findings, I show that, similar to physical eye-hand coordination, there is linearity of gaze-mouse correlation and gaze typically leads the mouse cursor movement with comparable lead time to eye-hand coordination. I also show a directional asymmetry of lead effect, i.e., leading distance varies if the proportion of horizontal components and vertical components of the trace changes; but the gaze-mouse tracing gain k in the horizontal or the vertical direction is consistent and irrelevant to the trace shape. In addition, the overall distribution of the distance between gaze and mouse cursor is constantly within a typical visual acuity range.

The dependency of gaze-mouse coordination demands in the tracing tasks was then addressed. I validated the hypothesis of a positive correlation between gaze-mouse coordination demands and gaze lead. Clearly, a task with higher coordination demands yields greater lead distance and greater lead time for gaze-mouse coordination. In scenarios such as tasks with extremely low gaze-mouse coordination demands, the mouse cursor led the gaze position in the horizontal direction. Yet neither was the tracing gain related to gaze-mouse coordination demand of the task, nor the overall distance distribution between the gaze and the mouse cursor. Finally, a new method of lead time calculation was proposed, which provided results without directional disparity.

In summary, the findings serve as a preliminary foundation for future research on the factors that affect causality between gaze and mouse cursor positions, which can also provide a theoretical basis for further improvement of gaze modulated input design, especially for indirect input devices and continuous manipulation tasks.

Chapter 4

Spatial Misperception

The previous chapter confirms that gaze leads when coordinating with mouse in continuous manipulation. As mouse and gesture control are both indirect inputs, it supports the idea to modulate gesture control using gaze lead, i.e., Gaze+gesture interaction technique (Chatterjee et al. 2015; Velloso et al. 2015).

I am interested in the Gaze+gesture technique due to the following reasons. Although both are indirect input methods, comparing to mouse, gesture control is a more natural interaction technique according to the NUI concept. It better reuses common human skills and it is more direct, for example, there is no need to convert a selection to a mouse click but simply grasping our palm. In addition, gaze facilitates fast acquisition because of gaze lead, and gesture control can provide richer control capabilities in all dimensions.

However, as gaze and gesture control use different sensors to track and detect user behaviours, the performance of the Gaze+gesture technique relies on users' perception on the exact spatial mapping between the virtual space and the physical space. An underexplored issue is when the spatial mapping differs with the users' perception, manipulation errors (e.g., out of boundary errors, proximity errors, see Figure 4.3 for an example) may occur. Therefore, in gaze modulated pointing, as gaze can introduce misalignment of the spatial mapping, it may lead to users' misperception of the virtual environment and consequently manipulation errors using gesture control (see detailed problem explanation in Figure 4.4). This chapter focuses on this issue and discusses its possible solutions.

4.1 Introduction

The working principle of the Gaze+gesture technique is illustrated in Figure 4.1 where the user uses eye gaze to locate an object and gestures to control the object in two different scenarios: (a) discrete actions and (b) continuous manipulation. Explanation of these two terms can be found in Section 2.3.1.

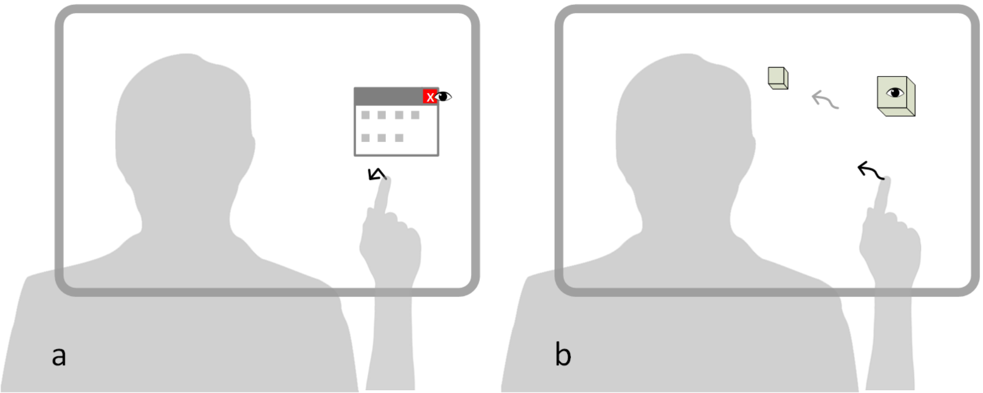


Figure 4.1: *Two examples of how the Gaze+gesture technique works in typical scenarios. (a) A 2D example of discrete action. To close the window, the user first looks at the "X" button on the upper right corner of the window and then makes a tapping gesture to close it. The user can tap their hand anywhere that is not necessarily on the button because gaze has located where the tapping will be effective. (b) A 3D example of continuous manipulation. To move the cube, the user first looks at it and then makes a dragging hand movement to move it. The user does not need to start to move their hand at the position where the cube was. The gaze decides which virtual object is going to be moved. The movement of the cube follows the hand movement in real time. The trajectory of the hand decides the cube's trajectory but starting from its own initial position. Note that the smaller cube after the movement indicates its movement in depth direction.*

Despite the advantages of the Gaze+gesture technique, Velloso et al. (2015) noticed that in certain scenarios as shown in Figure 4.1(b), it took users much longer time to select an object when they planned on subsequent manipulation on it after the selection. Based on their observation, they found that the issue was related to the position of the users' hand. That is, if the users' hand is not in an appropriate position that leaves enough room to manipulate the object after gaze selection, they must adjust their hand towards the object accordingly before picking it up. In other words, if the users pick up the object without clutching their hand towards it, the object will be picked up from an inappropriate position where there will be insufficient room for the subsequent manipulation. Clearly, the manipulation room depends on the tracking range of the depth sensor. If the users insistently manipulate within the limited tracking range, their hand will be lost in tracking once it moves beyond the tracking boundary, which will interrupt the continuous manipulation. Then the object will be dropped unexpectedly.

The interruption can appear in other forms. Surely there are many other ways to handle loss of tracking instead of simply dropping the object. For example, an intuitive way will be not dropping the object when tracking stops. This technique is applied in mouse control. When a mouse cursor hits the boundary of the monitor, it simply stays at the edge till the mouse moves backwards. However, with relative pointing devices (mouse), the physical space and the virtual space is always accurately mapped, so this method works. If we apply the same in gaze modulated gesture control, when the user's hand is lost in tracking, the object will hover in the virtual space long before it hits the virtual boundary and then resume movement when the user's hand comes back to the tracking area. The confusion or interruption will still exist as the user would not expect the object to stop moving while it is still in the middle of the virtual space.

This issue firstly compromises the faster target acquisition of gaze selection as users tend to clutch their hand close enough to the object to guarantee the manipulation room. Secondly, it harms the user experience as an unexpected interruption in a continuous manipulation can be frustrating. Additionally, why will there be insufficient manipulation room if a user tries to pick up an object far from it? This issue has not been previously reported in any other literature nor has it been further investigated by Velloso et al. (2015), so there is no clear explanation to help understand its causes. Given the fact that depth related 3D virtual object manipulation is very common in

VR and AR applications, this underexplored problem, although it occurs occasionally, cannot be ignored for two main reasons: (1) it compromises the faster target acquisition, and (2) it affects the user experience.

In addition, this issue was not found reported in any unimodal interactions using only mid-air gesture control either, so we argue that the issue is actually a spatial misperception problem which is related to the gaze modulated selection who introduces the misalignment of the sensor's spatial mapping.

In this chapter, the goal is to define the problem through a thorough investigation on its causes and specify the conditions of its occurrence, which are further testified through experiments. Moreover, three methods (Scaling, Magnet and Dual-gaze) is proposed for minimising the impact of the misperception problem, whose comparative performance and usability are examined.

4.2 Background

The spatial misperception problem related to user's gaze modulated 3D virtual object manipulation only occurs when specific conditions are met. These conditions involve the mapping techniques of the pointing devices, the input methods, and the types of the manipulation tasks. Before defining the problem, I first discuss whether existing interaction techniques have the risk of having this problem based on the three conditions.

4.2.1 Relative and absolute pointing devices

Many interactive techniques are integrated with gaze selection where different types of the pointing devices are involved. These devices need to map human behaviours from the physical space to the virtual space. Because a device has a tracking range and the virtual space is also limited, it is necessary to make sure to map the tracking area inside the virtual space. Depending on different mapping techniques used, the pointing devices can be categorised into relative pointing devices and absolute pointing devices (see Section 2.2 for details).

A computer mouse is a typical relative pointing device. Whenever the cursor hits the boundary of the virtual space, it will be restricted at the boundary even though the mouse keeps trying to push forward. At this time, the user can simply lift the mouse and relocate it to remap the relative position of the mouse and the cursor. In this case, the device will never lose tracking of the user's hand because their hand is always attached to the tracking device, even when it is integrated with gaze modulated pointing. As it is known that gaze improves the efficiency of selection by introducing a displacement from the location of the hand/tool cursor to the location of the gaze. Because relative pointing devices can be relocated or remapped when the displacement is generated, there will be no displacement introduced.

Some interaction techniques have the tracking devices or wearable sensors attached to the users' hand or body and so they do not rely on camera-based sensors to map the device into the world frame of the virtual space. These devices can also be considered as relative pointing devices. For example, Pouke et al. (2012) attached the sensor to users' hand to perform gesture control. Similar to a computer mouse, the gesture detection area was always centred on the physical hand position, so it can be considered as if it had an unlimited tracking range, thus when a virtual object is moved to the edge of the virtual space, the user can keep proceeding forwards if the display/virtual space is extended.

Touch screen and certain depth sensors for gestural control are absolute pointing devices. If the hand goes outside of the tracking area, or its mapped cursor goes outside of the virtual space, the hand needs to return to the tracking area to maintain the visibility of its cursor or itself in the virtual space. With absolute pointing devices, whether the displacement should be noticed depends on the following two conditions stated in Section 4.2.2 and 4.2.3.

4.2.2 Direct and indirect input devices

Theoretically, the absolute mapping is one of the prerequisites for the spatial misperception problem, but with some input methods, the tracking boundary can be explicitly shown, which helps users accurately perceive the tracking boundary and avoid the problem unobtrusively. Depending on how data or commands are fed into a system, there are two types of the devices, direct input devices and indirect input devices (see details in Section 2.2).

A touch screen is a typical direct input device that allows users to visually and tactually perceive its boundary. Even if there is an offset generated by a gaze selection as in the Gaze-touch applications (Pfeuffer et al. 2014), the user will not make any manipulation that proceeds outside the screen, regardless of the fact that the touch screen is an absolute pointing device.

Touch screens have also been used as indirect input devices such as the external manipulation device in distant displays (Stellmach and Dachselt 2012, 2013). In this case, the boundary of the devices could still be perceived by the users' hands even without looking. Other than that, depth sensors, mice, and joysticks are typical indirect input devices. When not considering the relative pointing devices, such as the mice, we can find the tracking range of the indirect devices is not explicitly indicated. Furthermore, the displacement introduced by the gaze selection updates the physical transform implicitly so that it cannot be precisely adapted to the mental representation. The awareness of the boundary can be more trivial when the task is very demanding and requires user's constant attention, not to mention that the transform makes the indirect devices more cognitively demanding than the direct devices (Charness et al. 2004). However, even when a device features the absolute pointing and indirect input techniques, the occurrence of the misperception problem using this device has one last condition to satisfy, which is the type of the manipulation.

4.2.3 Discrete and continuous manipulation

In a gaze modulated multimodal interaction, after the target has been accurately selected, the following manipulation is typically manifested by the hands. Chatterjee et al. (2015) defined the gaze selection as the target acquisition phase and the hand manipulation as the target action phase. They categorised the manipulation in the target action phase into discrete actions and continuous manipulation which I have mentioned in Section 2.3.1.

Figure 4.1 (a) gives an example of a discrete action. Discrete actions have no temporal position changes so they are not sensitive to the positions where the hand movement takes place as long as it can be captured by the tracker, so it is also not sensitive to the displacement.

Depending on different system requirements, some frameworks only use gesture control for making discrete commands, such as push and pull for zooming in and out (Yoo et al. 2010), extending two fingers for toggling the switch of an infrared light (Hales et al.). Thus the tracking range will not be a problem for these implementations.

Figure 4.1 (b) gives an example of continuous manipulation. It always initiates and ends with a discrete hand action, respectively. Between the two discrete hand actions, there is hand clutching which is continuously coupled with the movement of the virtual target. We define the clutching movement from the initial position to the end position as the trajectory of the manipulation. The trajectory can be changed by many variations such as the initial position, the moving direction and the CD gain. With a displacement of the initial position but the same moving direction and CD gain, the trajectory keeps its shape but is shifted relative to the tracking range. If the shifted trajectory cannot maintain itself entirely inside the tracking area, the tracing of the trajectory will be cut by the tracking boundary and the spatial misperception problem kicks in.

Therefore, the background of the spatial misperception problem discussed in this chapter is limited to the absolute pointing devices using indirect input methods during continuous manipulation. After clarifying the premises of the problem, I explain how it occurs in the next section.

4.3 Spatial Misperception Problem

In this section, it is first reviewed that how the conventional Gaze+gesture technique works in an object drag-and-drop example, then I can give a typical case of how the object would at times drop against a user's intention when being dragged. The problem that causes the interruption is then defined.

4.3.1 Gaze+gesture interaction technique (Normal method)

I implemented a prototype similar to the Gaze+gesture interaction technique presented in previous work (Chatterjee et al. 2015), which is referred to as the Normal method in this thesis for easy reference later in the experiment. The only difference is that we rendered a virtual hand as a 3D cursor to represent the need for the use of gestures. The selection workflow is that the users first stare at the object they want to grab, and then make a grabbing gesture at anywhere inside the virtual space, which confirms the selected object and changes its status to "selected". In the meantime, the virtual hand will be animated to shift from the grabbing position to where the selected object is as if the user is reaching out to the object. However, the physical hand remains still during the virtual hand shift. The animation of the virtual hand shift was implemented by linear interpolation.

A displacement is generated between the graphical hand position and the detected hand position due to the animated shift. The displacement origin is recorded once the grabbing gesture has been made. This information is kept until a releasing gesture has been made to drop the object. Please see Figure 4.2 for an example of the selection phase of the Normal method. Because the selection manipulation does not require physical hand movement to approach to the object, it reduces the arm movement to prevent from arm fatigue.

After the object is picked up, the users can move it with their hand to anywhere inside the virtual space, and this is the translation manipulation. The user does not need to stare at the object during the virtual hand shifting and translation manipulation because it is already in the "selected" status. When an object is incorrectly selected, the users just simply unfold their hand to "unselect", and the object stays at its original position. The grab gesture can be replaced with any other gestures or even an action of pressing a button.

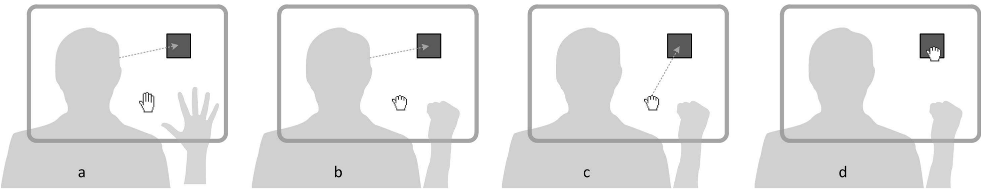


Figure 4.2: *Illustration of the selection phase of the Normal method. (a) The user is looking at the object to select with hand standby. (b) He makes the grabbing gesture to confirm the selection. (c) After the confirmation, the user does not need to stare at the object anymore. The graphic virtual hand shifts to where the object is, while the physical hand does not move. (d) The graphic virtual hand is shown grabbing the object.*

4.3.2 Interactive interruption

Interruptions were observed in interactions using the Normal method as illustrated in Figure 4.3, where the box indicates the tracking boundaries. As mentioned earlier, if the user's hand does not clutch towards the object before picking it up, the manipulative room will be restricted and thus potentially cause the problem of dropping the object. However, why will this happen?

In unimodal interactions with a gesture-only technique, the user knows where the tracking boundary is as the depth sensor is well mapped with the graphics. In that case, the user can gradually learn a spatial cognitive map of the tracking area in relation to their body through proprioception and visual displays (Jacobs and Schenk 2003) and maintain their performance as long as the physical mapping remains unaltered. However, when using multimodal interactions where several tracking devices are needed for supporting the interactions, all relevant devices have to be mapped with the virtual space properly and represented as a whole. A potential issue is that the mapping changed

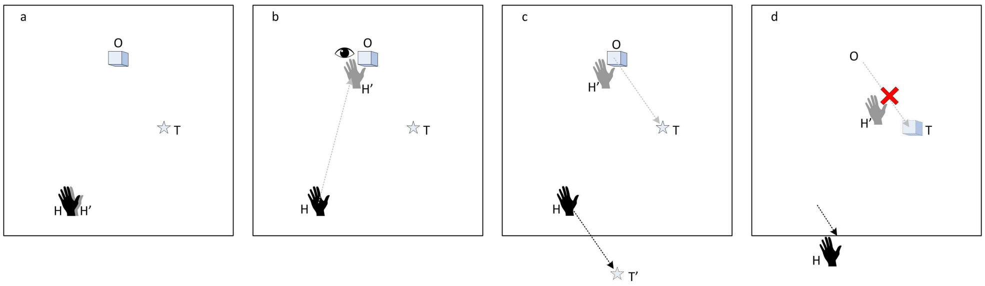


Figure 4.3: *A case of interrupted translation caused by mapping misperception. (a) The initial scene. The virtual space and the physical space are still aligned at this moment. The virtual hand H' and the mapped physical hand H are at the same position. (b) The user makes a grabbing gesture, and the virtual hand H' is warped to the cube at point O. The mapped physical hand does not follow H' but stays at its original position, thus creating a displacement between the two spaces. (c) The user plans to move the cube from position O to position T. To achieve that, the mapped physical hand H needs to move to point T'. (d) It is clear that the point T' is outside of the tracking area, so the movement of the mapped physical hand H can only be tracked till the boundary; the virtual hand H' can never get to its expected position T.*

by one modality can presumably affect the correct mapping of other modalities, causing that the tracking area of other modalities is no longer aligned with the original user perception. In the Normal method, gaze pointing improves the efficiency in selection by introducing a displacement between the virtual space and the physical space. This implicit displacement warps the mapping between the two spaces every time a new target is selected but the cognitive mapping could not catch up when there lacks a visual indicator. Thus, it causes misalignment of the physical tracking space and the visual display, i.e., the virtual space.

With this perceptual misalignment between the two spaces, the movements which are anticipated to occur inside the virtual space might go beyond the tracking boundary of the depth sensor. Without knowing of the potential interruptions, the user would perform the movement and get interrupted unexpectedly during the manipulation, which would impact their performance and frustrate them of using the system. As a result, it is important to find out under what conditions this problem will occur, and how to help users perceive the tracking range correctly, to inform the appropriate interaction design decisions to minimise or prevent such problems. Therefore, a definition of the spatial misperception problem is given in the next section.

4.3.3 Definition of the spatial misperception problem

In interactions using absolute pointing devices with indirect input methods during continuous manipulation, when the following condition is satisfied, the spatial misperception problem will occur and thus the hand will be lost in the sensor detection area:

Given a task that is to move an object at position O to a target position T, the distance from the object to the target is d, the moving direction is pointing from the object to the target. If a ray is generated on the moving direction from the position H where the hand picks up the object, it will eventually intersect with the detection boundary at a point I. The distance from the grabbing position H to the intersection point I is D. When d is greater than D ($d > D$), the spatial misperception problem will occur.

Figure 4.4 gives an illustration of the problem definition using the same example in Figure 4.3. Note that the problem defined here is different from the Out-of-Range (OOR) state described in the three-state model of input devices by Buxton (1990). OOR only describes a result, but here it explains a cause introduced by multimodal integration.

4.4 Strategies to Tackle the Problem

As clarified in the background section, the problem commonly occurs when the three conditions are satisfied in 3D manipulative tasks, absolute pointing, indirect input, and continuous manipulation. It should be noticed that the problem will not arise if any of the conditions is missing. In other words, as long as one of the three conditions can be removed in the design process, the problem will be resolved. The corresponding strategies are discussed below.

Firstly, recover the displacement during manipulation, or avoid generating the displacement. The relative pointing devices technically have no

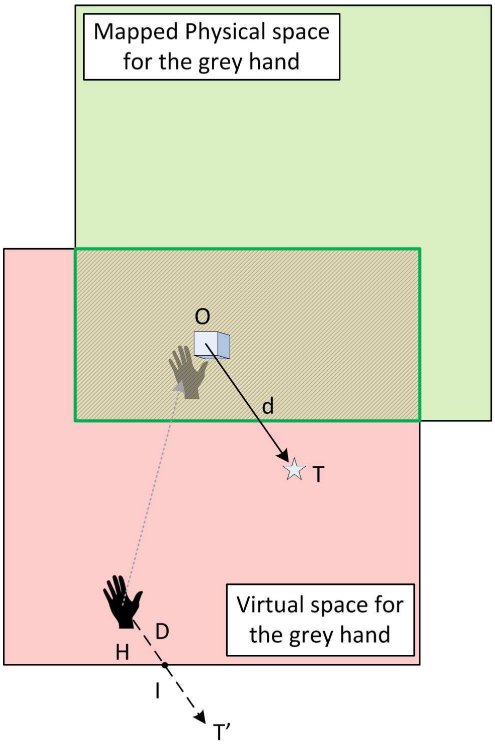

Mapped Physical space for the grey hand
O
d
T
D
H
I
T'
Virtual space for the grey hand

Figure 4.4: *An illustration of the problem definition. The dark hand represents the mapped physical hand in the virtual space; the grey hand represents the virtual hand (cursor). Before the displacement is generated, the mapped physical space equals to the virtual space which is the red zone. After the displacement is generated, the mapped physical space is also translated with the displacement to where indicated by the green zone. The valid working range for the virtual hand is restricted to the intersection of both zones which is indicated by the shaded area. The left and bottom boundaries of the intersection are provided by the depth sensor, and the other two are provided by the virtual environment. However, the latter are not necessary boundaries depending on how the virtual space is displayed.*

displacement generated, so in absolute pointing, to reduce the displacement, we can either decrease the CD gain to make the cursor moves faster so that a narrower physical workspace can still cover the whole virtual space; or let the user to adjust the initial picking up position subconsciously, i.e., to pick up as close as possible to the object. The shorter the displacement is, the larger the intersection of the two spaces is. For example, Frees et al. (2007) introduced an interaction technique that dynamically adjusting the CD gain to automatically recover the displacement without the users noticing it.

Secondly, use a virtual cursor to enhance the user's awareness of controllable boundaries. The direct input devices provide visible tracking boundaries but it is difficult for the indirect devices to do the same. However, a virtual cursor is helpful in this case. The virtual space is usually explicitly presented to the user, such as the border of the monitor. An intuitive mapping is to align the physical tracking area with the virtual space. With the help of a virtual cursor, a user can also "see" the tracking boundaries. Whenever the cursor disappeared in the virtual space, a boundary must be crossed. Virtual hand in gesture control can be considered as a 3D cursor. Many applications choose to explicitly display the virtual hand/cursor, such as Go-go (Poupyrev et al. 1996) and Homer (Bowman and Hodges 1997).

Thirdly, use discrete actions only to avoid interruptions in continuous manipulation. The discrete actions are not sensitive to the initial position of the gestural command, so it can be helpful to convert the continuous manipulation into a set of discrete actions. A drag-and-drop task can consist of a gestural command at the picking up position and another gesture at the dropping position, but it has limitations when the trajectory between the two positions needs to be traced accurately.

Based on the discussions above, three possible solutions are proposed: Scaling, Magnet and Dual-gaze where the first two are derived from the first strategy and the last can be seen as an example of the third strategy. Note that all solutions are incorporated with a virtual hand as a virtual cursor as suggested in the second strategy.

4.4.1 Gaze+gesture with scaling (Scaling method)

This method, referred to as the Scaling method for easy referencing, represents the strategy that recovers the displacement imperceptibly. This method supports the same manipulation style as it is supported in the Normal method but its translation stage is rendered differently from the latter, which distinguishes the two methods. In the Scaling method, the translation will be scaled proportionally according to the relative position of the virtual hand and the boundary when the system detects the current moving direction is likely to cause user's spatial misperception.

Specifically, two rays will be generated, one in the instantaneous translation direction OT from the object O, and another in the same direction from the detected hand position H (Figure 4.5). Remember this detected hand is invisible and it is different with the graphical virtual hand which can be seen grabbing the object. The displacement HO represents the difference between the two hand positions. The first ray gives the distance D_o from the object to the boundary in the translation direction. The second ray gives the distance D_h from the detected hand position to the boundary in the same direction. When $D_o <= D_h$, nothing changes; when $D_o > D_h$, the scaling scheme is applied, i.e., we obtain the real-time hand translation difference Δd between this frame and the last frame, and then calculate its proportion on $D_h(\Delta d/D_h)$ and multiply it with D_o to get a proportional distance s that the graphical hand needs to move.

$$s = \frac{\Delta d}{D_h} \cdot D_o \tag{4.1}$$

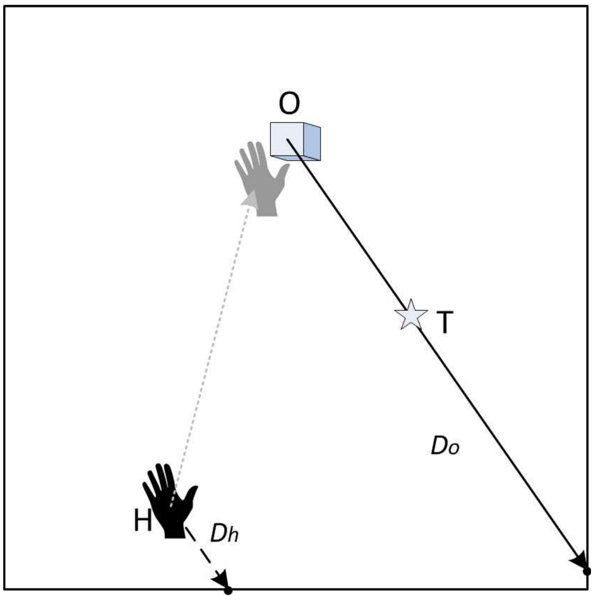


Figure 4.5: *Illustration of the Scaling scheme.*

Note that D_h is the same as D in the problem definition (Figure 4.4), which is the distance from the detected hand position to the boundary; but D_o is different with d, that D_o is the distance between the object to the boundary and d is the distance also from the object but to the target. This is because the target position is unknown when the hand starts to move, we pick the proximity to replace the unknown value here. The scaling scheme can make sure the hand never goes beyond the detection boundary. Please see Appendix A for more details of the scaling scheme.

4.4.2 Gaze+gesture with magnet (Magnet method)

This method, referred to as the Magnet method for easy referencing, represents the strategy that converts the absolute pointing to relative pointing for not generating displacement. This method uses a metaphor that the hand is magnetic, like using a magnet to collect metal objects. It differs from the Normal method in the selection stage (Figure 4.6). Although the manipulation workflow is the same (i.e., the user looks at an object and makes a grabbing gesture), the graphical virtual hand does not shift to where the object is, and the object, instead, is attracted to the virtual hand. The following translation manipulation is the same with the Normal method. When an object is incorrectly selected, the users can open their palm to unselect, and the object will drop at the current position. Kitamura et al. (1998) used a similar magnetic metaphor but they applied it on the objects instead of the virtual hand/tool.

This method also guarantees all the movements are inside the detection boundary because there is no displacement between the graphical virtual hand and the mapped physical hand. It is achieved by changing the object's position instead of the virtual hand's position.

4.4.3 Gaze+gesture with dual-gaze (Dual-gaze method)

This method, referred to as the Dual-gaze method for easy referencing, represents the approaches that convert the continuous manipulation to a set of discrete actions. As the name suggests, the functionality of gaze is extended to unselecting objects as well in this method. It follows the interaction flow described by Turner et al. (2013): object location, confirmation of selection,

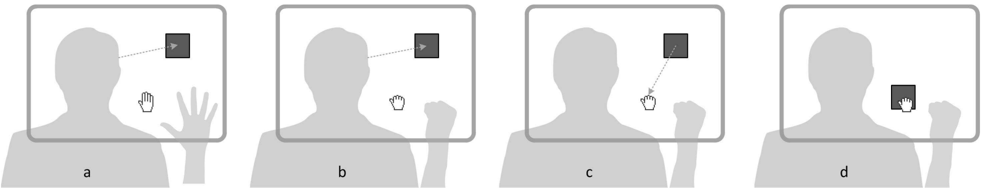


Figure 4.6: *Illustration of the selection phase of the Magnet method. (a) The user is looking at the object to select with hand standby. (b) He makes the grabbing gesture to confirm the selection. (c) After the confirmation, the user does not need to stare at the object anymore. The object shifts to the location of the graphic virtual hand, while the physical hand does not move. (d) The graphic virtual hand is shown grabbing the object.*

destination location, and confirmation of dropping. The *locate* attribute is fulfilled by the gaze, and the *confirm* attribute is fulfilled by the gesture. This method differs from the Normal method in the translation stage (Figure 4.7). Other methods all require users to physically move their hand in order to move the object to the target position. In this method, a user does not need to move their hand at all. After the object is picked up by the user, by simply looking at the target and making a release gesture, the object can be translated to the target automatically. Linear interpolation is applied to the virtual hand movement during the animated translation.

Even though this method keeps the displacement between the virtual space and the detection space, and the displacement will be updated once the release command is done, it still avoids the spatial misperception problem by replacing the continuous translation with a discrete gesture command.

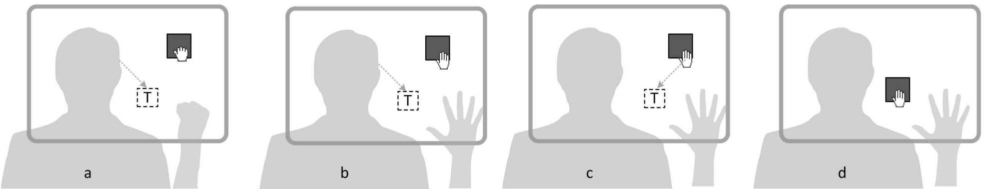


Figure 4.7: *Illustration of the translation phase of the Dual-gaze method. (a) After the object is selected and grabbed by the virtual hand, the user is looking at the target position. (b) The user releases their hand to confirm dropping the object to the position where they are looking at. (c) After the confirmation, the user does not need to stare at the target position anymore. The virtual hand and the object shift together to the target position. (d) The graphic virtual hand is shown released the object at the target position.*

4.5 Experiment

The aims of the experiment are: (1) to validate the problem defined in Section 4.3.3, and (2) to testify whether the three proposed methods can resolve the problem through usability measurements. Thus, the hypotheses based on the aims are:

- when using the Normal method, the problem will occur if the problem condition is met;
- when using the Normal method, the problem will not occur if the problem condition is not met;
- when using any of the three proposed methods, the problem will not occur no matter if the problem condition is met or not.

In order to test the hypotheses, two task scenarios showing common usages were created: S1 (drag and drop a single object) and S2 (drag and drop multiple objects). Please see Figure 4.8 for an illustration of the two scenarios. The main purpose of S1 was to validate the problem definition. To achieve this purpose, the task was simplified in a strictly controlled environment where both grabbing and target positions are fixed so that the problem condition could be easily reproduced by only changing the cube's position. Only one object was tested in each trial under two conditions which were deliberately setup:

- OUT condition is where the manipulation would go out of tracking boundary based on the problem definition.
- IN condition refers to the condition that does not follow the problem definition where the manipulation would stay inside of the tracking range.

In S2, we wanted to test if the defined problem would happen when the IN and OUT conditions were not controlled. This is because the grabbing position, the target position and the object position could not be controlled in real applications where the problem may not occur at all purely based

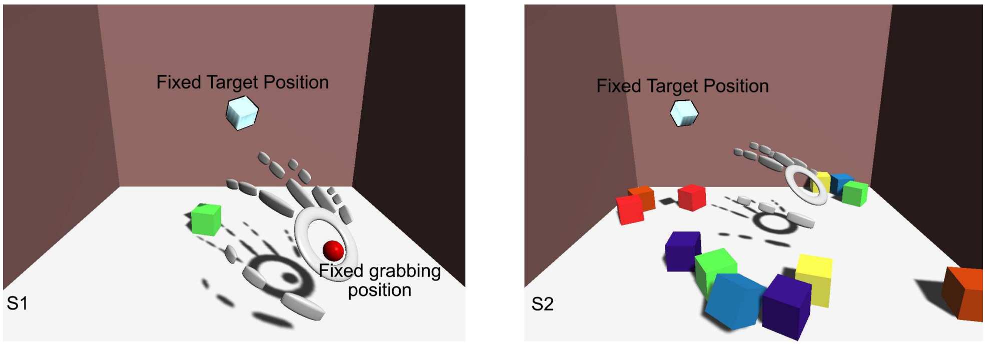


Figure 4.8: *Illustration of S1 and S2. In S1, the red dot indicates the fixed grabbing position. It only turns red when the virtual hand overlays with it, otherwise, it is grey. The participant can only start the trial when the dot turns red, i.e., the participant should always start to move their hand from the dots position. In both scenarios, the highlighted floating object indicates the fixed target position where to drop the cubes.*

on the users' interactive habits. Therefore, a more general task with multiple randomised objects was tested. With only the target position fixed, the participants obtained full control flexibility to avoid the problem. However, although it was possible that all objects were picked up without the misperception problem risk and vice versa, it generally should be a mix of both conditions as the participants would not proactively avoid the problem because they were not aware of such problem and when it would occur. The purpose here was no longer testing if the problem condition but whether it could be triggered in real interactive environments as opposed to unrealistic experimental environments in S1.

The task completion time, errors and user preference were tested in both scenarios for the usability comparison study.

4.5.1 Apparatus

Participants sat $66cm$ away from a desktop running the experiment built with the Unity game engine. A 23" HP Compaq LA2306 LCD monitor featuring Full HD 1920×1080 resolution with the refresh rate at $60Hz$ was used as the display in the experiment. Tobii EyeX was used as the eye tracker mounted to the bottom edge of the display with estimated 0.4 degrees of visual angle accuracy and the sampling rate used was $60Hz$. The viewing was binocular and the calibration was conducted with both eyes. The participants' hand movement was tracked by a Leap Motion sensor placed facing up on the desk about $45cm$ away from the display. The size of the virtual space was automatically generated based on the tracking space. The SDK for gesture recognition was provided by Leap Motion whose recognition accuracy could achieve 89.3% for the grabbing gesture and 97.1% for the releasing gesture according to Marin et al. (2015). The eye tracker, the motion sensor and the display were set up as shown in Figure 4.9.

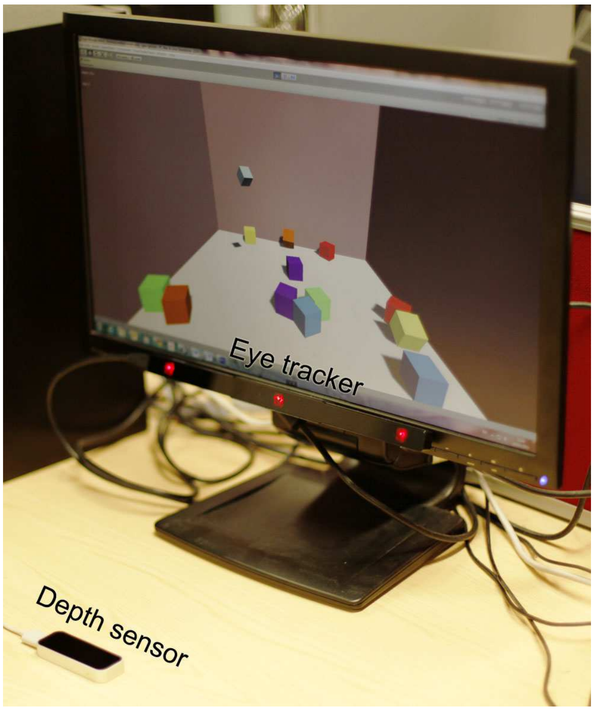


Figure 4.9: *Experiment setting up.*

4.5.2 Participants

Twenty participants, 12 male and 8 female, aged between 23 and 41 (Mean$\pm$ SD = 27.5 ± 4.2), volunteered themselves in the study. None of them had any eye movement, hand movement or neurological abnormalities. They either had adequate natural visual acuity or corrected vision with glasses. Except for one participant, others all reported being right-handed. Written consent was obtained from each of them after explanation of the experiment. Before starting the tasks, participants were asked to answer some background questions by rating a $5-$point Likert scale from 1 Strongly disagree to 5 Strongly agree. All the participants stated that they mainly used mouse and keyboard for computer interaction (Mean±SD = 4.9 ± 0.3). Most participants never used mid-air gesture control except for seven participants (Mean±SD = 1.8 ± 1.2). As for using eye tracker as an interaction interface with computers, only four reported they had some experiences (Mean±SD = 1.4 ± 0.8).

4.5.3 Procedure

The user study started with a brief introduction followed by a demographic questionnaire as described in previous Section 4.5.2. Please refer to Figure B.1 in Appendix B for a screenshot of the demographic questionnaire. The participants were instructed to sit fairly still without restricting their movements especially head movements. Before practising each method, a 9-point grid calibration was performed. Then one method at a time was described to the participants and the participants were asked to practise the method until they felt confident. Their performance was recorded after they had practised all four methods and confirmed they were ready to start the formal tests.

In both scenarios S1 and S2, the participants were asked to grab and move a cube or cubes to the target position. The target position was marked by a referencing object, once the cube collided with the target object, the cube itself would disappear, indicating a successful trial. Each method was tested as a group but the order of the four groups was randomised. As the problem discussed in this chapter is position related, the orientation of the objects has little impact on the problem definition. In order to remove possible variation caused by orientation change, it was restricted to be 3-DOF in the task implementation, so that the selected object could not be rotated with the hand orientation. Thus, the selected object kept its original orientation in all circumstances unless it collided with the physics-enabled environment.

In S1, each participant was asked to perform 5 trials under each condition (IN and OUT) per method (5 trials × 2 conditions × 4 methods = 40 runs). The order of the 10 trials in each method was randomised.

In S2, one task block contained twelve cubes. Three task blocks were tested for each method (3 blocks × 4 methods = 12 runs).

After each block of a method was completed in S2, the user was given a SUS (System Usability Scale) (Brooke 1996a) questionnaire to complete. Please refer to Figure B.2 in Appendix B for a screenshot of the SUS questionnaire. A post-task interview was also conducted to collect qualitative feedback.

4.5.4 Measures

The quantitative evaluation included three parts: the task completion time, the error rate or error count, and the SUS score. The qualitative evaluation included a post-task interview asking for feedback on the overall experience on what the participant liked and disliked in each of the tested methods to help us understand their preference.

Task completion time Task completion time was defined as the time a participant spent to complete a task trial using a method in a specific scenario. For S1, the timer started as soon as the cube was selected and stopped as soon as the cube disappeared. For S2, the timer started when the first cube was selected and stopped when the last cube disappeared.

Error rate / error count Error rate was used in S1 and error count was used in S2. In S1, if the participant moves their hand out of the detection area in the middle of translating a selected cube, their hand will be lost in tracking and the cube will be dropped unexpectedly before reaching the target position. This will be counted as an error, indicating an occurrence of the misperception problem. Each trial in S1 had only one cube tested, so as long as the cube was dropped once in a trial, the trial was counted as an error trial. Therefore, an error rate can be obtained according to the proportion of the error trials among the whole trial set. In S2, the error count increases every time a cube is dropped in one test block. No error rate was calculated for S2. Note the loss of tracking error and accidentally dropping error are not distinguished here.

SUS score The SUS (Brooke 1996a) was presented with a ten-question questionnaire with a $5-$point Likert scale from 1 Strongly disagree to 5 Strongly agree. Note that the questions ordered with an odd number are positive statements of the system and the even numbered questions are negative statements of the system. Please refer to Appendix B for details of the questionnaire. A $0 - 100$ score can be calculated from the ten ratings as a numeric evaluation of subjective assessment. To obtain the SUS score, the $1 - 5$ ratings were firstly normalised to $0 - 4$ where the contribution from the odd questions was the rating minus 1, and the contribution from the even questions was 5 minus the rating. It guarantees that high rating always indicates positive evaluation. Then the sum of the ratings was multiplied by 2.5 to yield the final score. In practice, the average SUS score is 68, indicating 50% preference (Sauro 2013).

4.6 Results

The task completion time and error rate give a clear indication of the system performance, so as the questionnaire to the usability. A one-way ANOVA was used to investigate the differences among the four methods in task completion times both in S1 and S2. Post hoc comparisons using the Tukey HSD test were performed to further identify which method was significantly different from the others.

4.6.1 Completion time

Figure 4.10 shows the completion time for each method under the IN and OUT conditions. The one-way analysis of variance revealed significant differences between these four methods in both conditions (IN: $F(3, 396) = 24.29$, $p < .0001$; OUT: $F(3, 396) = 124.2$, $p < .0001$). It is noticed that participants took a longer time to complete tasks using the Normal method and the Scaling method in the OUT condition. For the Normal method, it is because, in the OUT condition, the object was prone to drop, it cost more time to pick it up and move it to the target again. In the Scaling method, the scaling could prevent dropping the object which saved time, but it was not very smooth and it tended to overshoot when the participant moved the object with a high speed. When this occurred the participant needed to move the object back from the overshoot position and hence cost more time. Both of the Magnet and Dual-gaze methods could help the participants achieve equally short completion time regardless of which condition, showing that the conditions have no impact on these two methods. The reason why the completion time was shorter within Magnet and Dual-gaze in the IN condition could be that they required less arm movement than other methods. In short, the results indicate that the Normal method requires more time in the OUT condition; the proposed methods can reduce the completion time in the OUT condition to different extents; and that all techniques require less time in completing tasks under the IN condition.

Figure 4.11 shows the overall completion time for each method in S2. Because there was no constraint of the initial hand position in this task, the conditions were mixed. Thus, this figure demonstrates the occurrence of the defined problem in more general cases. The one-way ANOVA yields a significant difference between the four methods, $F(3, 236) = 55.26, p < .0001$. The post hoc test shows a similar result to what was discussed regarding Figure 4.10, that the scaling scheme shows a little improvement in efficiency but not quite as much as the Magnet and Dual-gaze method. Again, the participants performed better using the last two methods in terms of completion

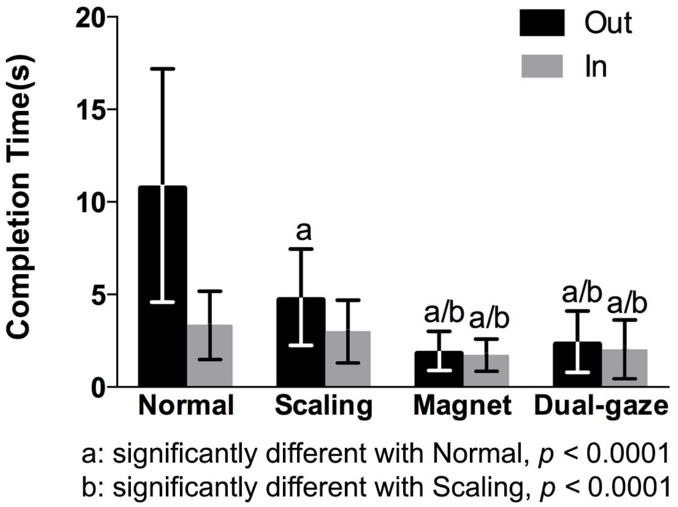


Figure 4.10: *Completion time for each method under the two conditions in S1. Error bar indicates the standard deviation.*

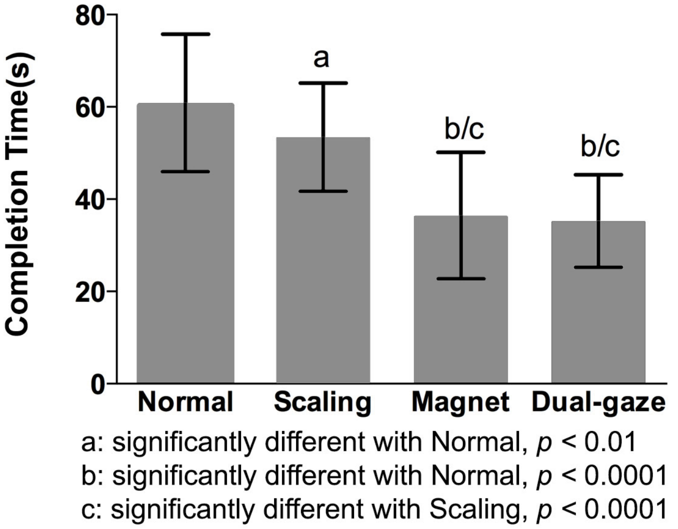


Figure 4.11: *Completion time for each method in S2. Error bar indicates the standard deviation.*

time. This result also shows that the OUT condition still has a high potential to occur when the environment is not deliberately setup, which supports our assumption that the reason why participants performed worse with the Normal method was due to the OUT condition.

If we define the efficiency as the average time cost by drag-and-dropping one cube, we can find that the Scaling method improved the efficiency by 11.59% compared to the Normal method, the Magnet method improved 39.99%, and the Dual-gaze method improved 41.80%.

Method	S1 Error Rate		S2 Error Count
	IN	OUT	
Normal	0.03	0.99	173
Scaling	0.01	0.14	82
Magnet	0.02	0.03	35
Dual-gaze	0.04	0.05	39

Table 4.1: *Error rate for S1 and error count for S2.*

4.6.2 Error rate

Table 4.1 gives a summary of the error rate for each method under different conditions in S1, as well as the total number of errors occurred in S2 for each method respectively. There is a positive correlation between the completion time and the error rate/count. That is, the longer time a participant took to complete a task, the higher error rate they will end up with or the more errors they will make. Typically, the error rate of the OUT condition of the Normal method has reached 99%, which supports our problem definition. The 1% trials that should produce errors but none in the actual experiments were caused by the object bouncing. Because the virtual environment was implemented with physics, when an object was released, it collided with the wall and bounced to a position that perfectly avoided the OUT condition.

Note that there are still some errors recorded when it is expected no error should happen, such as the IN condition for all methods, and the OUT condition for Scaling, Magnet and Dual-gaze. These error are accidental drops, mainly caused by the instability of the hand tracking. This instability was caused by the interference of the eye tracker as both trackers used infrared light for detection. The eye tracker was mounted higher than the hand tracker, so its light would interfere with the image caught by the hand tracker, and made the image flicker. This issue became significant when a participant lifted their hand to the height of the eye tracker.

In Magnet and Dual-gaze, judging that there is no significant difference between the error rates of the IN and OUT conditions, and the error rates are very low, we may assume the errors are caused by accidental drop which happens in both conditions, and it does not impact on our hypothesis testing. The

Method	Mean	SD	Min	Max
Normal	69.4	16.2	32.5	92.5
Scaling	67.9	16.9	32.5	95
Magnet	87.9	9.2	72.5	100
Dual-gaze	85.9	12.5	62.5	100

Table 4.2: *SUS score for each method.*

error rate of the OUT condition in the Scaling method is lightly higher because of the overshoot. The same explains the error count differences among the four methods in S2.

4.6.3 Preference

Table 4.2 shows an overview of the SUS score for each method. The range of a SUS score is between 0 and 100 from low to high satisfactory. As expected the last two methods scored much higher than the Normal method. Surprisingly, Scaling scored the lowest. According to the post-task interview, sudden acceleration and overshoot were not as tolerable as losing detection or dropping the object. Some participants complained about eyes getting tired during the dual-gaze tasks, which could possibly explain why the score for the Dual-gaze method is slightly lower than the Magnet method.

The SUS score breakdowns shown in Figure 4.12 were obtained from the normalised ratings that range from 0 to 4 (the normalisation was explained in Section 4.5.4), so high ratings always indicate positive evaluation. The Magnet and Dual-gaze methods outperformed the Normal and Scaling methods in almost all of the questions, only in question 10 that compared to the Normal method, the proposed methods showed the requirement of a longer learning curve.

The scores for Magnet and Dual-gaze were very close to each other. Only in question 2, 4, 6, and 10, the Magnet method was rated higher than the other. As these questions are related to the complexity and learnability of the system, it indicates that the Dual-gaze method was not as natural and easy to learn as the Magnet method. Similarly, the Scaling method had very close ratings to the Normal method but the difference in question 7 and 10 indicated the Scaling method was more complex and difficult to learn than the Normal method.

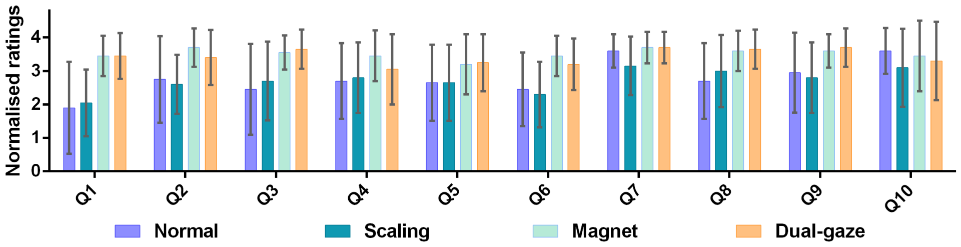


Figure 4.12: *SUS ratings breakdown for each method. Error bar indicates the standard deviation.*

4.7 Discussion

The results confirm that the unexpected dropping was caused by the problem defined in Section 4.3.3. Furthermore, the proposed three methods provided circumstantial evidence that by removing some of the sufficient conditions as discussed in the background (Section 4.2) and Section 4.4, the problem no longer existed.

4.7.1 Advantages and limitations

Overall, the results reveal that the scaling scheme improves the performance of the Normal method but it is still sensitive to the OUT condition. The Magnet and the Dual-gaze methods are tolerable to the OUT condition and the two have comparable performance. In other words, the Scaling method recovers the displacement gradually, but the Magnet and the Dual-gaze methods have no displacement generated at all. It indicates that the participants can perceive the mapping change, but a consistent mapping benefits the user experience.

The advantage of the Scaling method is that it alleviates the possibility of interrupted translation when the tracking space is not enough for isometric movement. It is also easy to learn because the manipulation is identical to the Normal method, which requires no necessity of training before use. However, the usability improved by the scaling scheme is counteracted by the lack of smoothing which makes the participants aware of the scaling but not aware of when it will kick in due to the manipulative similarity to the Normal method. Introducing a hidden affordance cannot perfectly solve the false affordance problem in this case. The user experience evaluation has shown that the participants were not very satisfied with the way it scaled, thus a smoothing adjustment of the dynamic CD gain is expected to be integrated into this method.

The advantages of the Magnet method are its stability and efficiency. It outperformed all other methods in this study. The low error rate contributed to its stability because the interruption rarely occurred, meanwhile, its short task completion time and requirement of low arm effort assured its efficiency. Although the translation stage still requires physical hand movement, for tasks like S2, when several objects need to be moved to the same target or targets close to each other, it will be convenient to keep the hand at one position to attract the objects to the vicinity of the target and keep the physical hand movement to the minimum.

Similar to the Magnet method, the Dual-gaze method also showed good performance in terms of efficiency and stability. It requires the minimal effort from the hand and arm but it does require effort from the eyes which may lead to fatigue, especially when using the eyes again to locate the dropping position. No tiredness of eyes was reported in other methods that only required using gaze once to select. Many participants were fascinated with this novel interaction paradigm and preferred this method even when the Magnet method was less prone to eye fatigue. Both Magnet and Dual-gaze methods can avoid generating spatial displacement, but the Magnet method requires less eye effort and it encourages the users to adjust the initial picking position to reduce arm movement. The extra advantages make the Magnet method a better interaction technique in gaze modulated gestural control.

A limitation of the Dual-gaze method is the requisite of target awareness which needs a known target position to move to. In my implementation, I used a referencing object to indicate the dropping position. However, not all the manipulations will have a known target, so in these manipulations, the Dual-gaze interaction technique is not suitable. The target object also provided depth information for the gaze selection in 3D because eye trackers could only provide 2D positional information. As a result, 3D target acquisition is restricted by the gaze selection. It is possible to extend gaze selection from 2D to 3D if more than one gaze point can be obtained for the same target, where the depth can be calculated similarly to the vergence eye movement. Alt et al. (2014) proposed a method that using the ocular vergence to determine the gaze position in 3D space by measuring the distance between both eye pupils, as pupils would rotate inwards simultaneously when looking at close objects and vice versa. They also proposed another method by gauging the pupil diameter as it changes according to the distance of the intended object. Instead of extending gaze pointing to 3D, a multimodal solution that takes advantage of the 3D accessibility of the gesture control can be applied to determine the depth of the target position in the absence of a referencing object.

A limitation of the experiment is that the object distribution was always on the ground because of the involvement of the gravity for simulating a physics-enabled environment. This has constrained the movement of the objects as they had to be moved upwards in most cases. This condition should be removed so that we can test a truly random distribution where no external forces are involved as in outer space.

Some refinement should be noted for the proposed techniques, e.g., improve the smoothing of the CD gain recovering of the Scaling method, enable 6-DOF manipulation, and develop depth acquisition on top of gaze modulations to broaden the usefulness of the Dual-gaze method.

In summary, the proposed approaches have improved the usability to different extents, but because they were only developed as prototypes for concept demonstrations, more features to perfect these methods need to be considered and implemented. Moreover, the results revealed that an interaction technique with consistent spatial mapping and lowest fatigue was preferred in continuous 3D manipulation.

4.7.2 Design recommendations

Not only can camera-based eye tracking and depth sensors, but other devices which are absolutely mapped indirect input devices, benefit from this study when they are dedicated to continuous manipulation. For example, stylus-based haptic devices track the body movement using links and joints instead of tracking sensors, which also have a tracking area with boundaries that is restricted by the kinematic workspace. Furthermore, even if the detection area or workspace is wide enough to cover all the possible movements, a human arm still has a limited reach itself meaning that a more constrained space will be still formed regardless of the coverage of the actual detection area and workspace. In this case, the mismatch will be extended to the mapping between the virtual space and the physical arms' reach.

Although all proposed approaches can generally be alternatives to each other, they still can be used to support specific tasks due to their speciality. The Magnet method is suitable for repetitive picking-up when the targets are located very close to each other. For example, when building a LEGO model, the users can rest their hands near the model and pick up building bricks by gaze. The selected bricks will fly to their hands and they only need to move a small distance to the expected position. The Dual-gaze technique has great potential to help motor impaired users as it only requires minimum arm movement. However, both methods cannot provide a designated moving path for the objects, in which case the Scaling method can be applied.

When multiple trackers are adopted into the interaction, at most one infrared light facilitated tracker is recommended, otherwise the trackers should be deliberately positioned to avoid light interference. Common desktop mounted or display mounted eye trackers are using infrared light. Modern depth tracking sensors also use infrared light. If multiple light sources interfere with each other, it will reduce the stability and accuracy of all infrared trackers, i.e., the hand tracking and eye tracking devices in this study. This issue did not affect our results because all methods had this problem and it was counterbalanced in the comparison. Furthermore, there should be no such issues in eye tracking and gesture control enabled VR headsets, because the eye tracker component is placed inside the headset and the depth sensor for hand tracking is outside, which perfectly avoids light interference. However, interface designers should bear this interference in mind.

High-precision eye trackers or run-time recalibration is recommended. Although in our experiment, the accuracy of the eye tracker was satisfactory, there were still circumstances that the participants were trying to pick up an object occluded by several other objects. I applied an eye-slaved zoom lens similar to what was developed by Stellmach and Dachselt (2012) to solve the partially occluded problem, but there are many other alternative solutions for selections with partial and even full occlusions as I reviewed in Section 2.4. However, no particular solutions of the ambiguity problem in the Gaze+gesture background has been reported yet, which I will discuss in the next chapter. Preferably, the eye trackers should evolve to provide higher precision and calibration accuracy but still remain cost effective. Alternatively, some interface does not rely on accurate calibration to determine the object of interest by dynamically correlating eye pursuit movement to the movement of a moving object (Vidal et al. 2013), but this method is not suitable for statically displayed scenes. Apart from the occlusions caused by the virtual objects, the hands and arms of the users can also cause occlusions between the eyes and the display. Such a problem can be well-controlled using indirect input (Simeone and Gellerseny 2015), so the gaze modulated techniques are capable of addressing the hand occlusion problem due to its indirect feature.

4.8 Summary

Multimodal interaction, on the one hand, integrates advantages of each modality for a greater combined usability, on the other hand, it can introduce new problems that do not exist in either unimodal interactions. The spatial misperception problem discussed in this chapter is one of the problems caused by multimodal integration.

This chapter identifies this problem and contributes to enriching the design guidance for multimodal interfaces of 3D manipulations based on eye tracking and mid-air gesture control. To the author's knowledge, it is the first study identifying the spatial misperception problem, laying out the theoretical foundations for further engineering and experimentation.

Chapter 5

Disambiguation

Changing one feature of an interaction design may impact every other aspect of the original design, such as ambiguity scenarios handling. Considering the synergy of NUI designs, designing interactions in a natural way may require an adaptive technique to naturally handle ambiguities too.

As briefly aforementioned, there are no solutions particularly designed for the ambiguity problem caused by partial or full occlusion in the Gaze + gesture technique yet, despite of the fact that many disambiguation techniques are developed for traditional interaction techniques. In addition, neither can eye tracking techniques nor the virtual hand metaphor achieve comparable pixel-level accuracy to mice. This may cause problems when fine grained interactions are required, such as selecting in a dense virtual scene where objects are close to each other and occlusion is prone to occur.

In order to solve this problem but also adapt the solution to the Gaze + gesture design, a coarse-to-fine two-step disambiguation technique is proposed in this chapter. Specifically, the reflexive gaze shifts are embedded with the *locate-and-confirm* two-step paradigm of gaze modulated pointing, which is novel to use gaze variants for solving ambiguity problems in gesture controls.

5.1 Introduction

Eye trackers can hardly achieve pixel-level accuracy due to two reasons. First of all, the algorithm that maps the captured eyes image to a point on the screen typically delivers some error ranging from 0.5 to $1.7cm$ wide (Monden et al. 2005). Moreover, the physiological nature of our eyes constantly introduces tremor and natural random offsets (Špakov 2011). It seems not a big problem when selecting in a virtual environment that is sparsely distributed with objects of decent sizes, but it can degrade selection efficiency in dense and occluded virtual scenes. I modified Figure 2.4 to show how gaze pointing introduces selection errors under these object interaction scenarios as shown in Figure 5.1. In Figure 5.1 (a) and (b), the two objects are proximate to each other, even partially occluded. If the purpose is to select the blue cube on the left and the user is already looking at it, the corresponding gaze point can still be mapped with a small offset to its neighbour on the right, and thus a selection error may occur. I have defined these two cases as the proximity problem in Section 2.4.

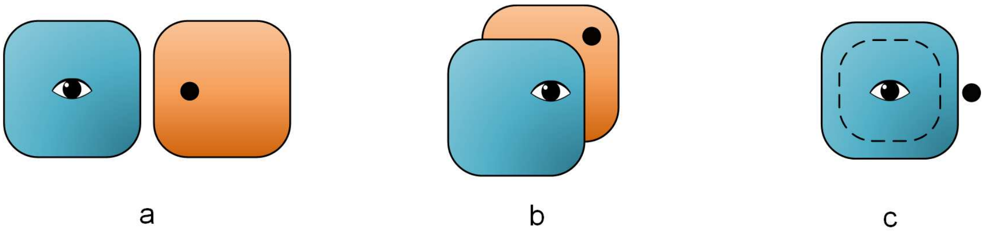


Figure 5.1: *Scenarios that gaze selection may be prone to errors in 3D interaction. The eye indicates where the user is actually looking while the dot represents where the eye tracker thinks the user is looking. Note there is an offset between the eye and the dot. (a)Proximity. (b)Partial occlusion. (c)Full occlusion. The dashed outline represents an object behind the blue cube.*

The third scenario that not only requires accuracy but also accessibility to the depth is to select a fully occluded object. As shown in Figure 5.1 (c), the target is perfectly occluded by the front blue cube, which is defined as the occlusion problem in Section 2.4. Ray-casting normally returns the front object instead of the object behind, or a set of candidates intersected by the ray for further disambiguation. It still requires accurate pointing on the exact position where the ray can go through the target and accessing the depth in the meantime. Thus, a solution is needed to tackle both the planar offset and depth accessibility, which takes advantage of the features of gaze modulated techniques and gestural control without deteriorating the usability.

The proximity and occlusion problems were investigated previously using unimodal inputs such as game controllers (Cashion et al. 2012). As the Gaze+gesture technique has just recently emerged, no particular discussion is reported regarding whether the existing solutions still suit the new technique. Thus, this chapter aims to present how disambiguation techniques are adapted and developed in the Gaze+gesture technique, as well as a comparative study to understand the usability of this proposed technique.

5.2 Design of Gaze Modulated Disambiguation

According to related work (Section 2.4.1), volumetric probes and distortion can both solve the proximity problem and occlusion problem. Considering the consistency of interactions, it is desired to use the same interactive pattern under all circumstances, which can reduce user's learning time and confusion during interactions (Mandel 1997). Among the two, distortion is preferred by global tasks because it provides more context information while volumetric probe deals with local scope. Considering gaze is a natural local filter, we design a volumetric probe using a gaze cone and a gaze probe to solve both proximity and occlusion problems in this chapter.

Here I describe the details of how the proximity and occlusion problems are solved using the multimodal features of gaze and gesture. This design follows the coarse-to-fine selection patterns with the two steps: ambiguity detection and decluttering. A state transition graph is also illustrated to provide a case of how to integrate this disambiguation technique into a fluent selection and manipulation flow.

5.2.1 Conical ambiguity detection

A right circular gaze cone (Forsberg et al. 1996) is applied, which is invisible to the users to realise volume selection, so small targets and missed targets caused by the gaze mapping offset are captured. Figure 5.2 shows an example of a gaze cone. The height of the cone should be long enough to reach the far clipping plane of the camera. Typically, the cone is always centred with the user's gaze ray, so from the user's perspective, the cone always looks like a circle, as shown in Figure 5.2a. We can define the size of the cone using the diameter of the circle. The distance between the screen and the user usually remains in a limited range, if we treat it as a fixed value, we can also set the size of the cone to a defined value. If the size of the cone is too large, too many objects will be included so the filtering effect is not significant. If the size is too small, there will be little difference with the ray-casting selection and our problem remains unsolved. Here the size of the cone is set about 5° of visual angle. As I have explained in Section 3.3.4, the visual acuity area extends about 10° around the centre of the retina, so 5° is the median size of the 10° visual acuity angle. It can tolerate up to 2.5° eye tracking errors.

An object is considered to be inside the cone if its centroid is inside. If more than one object is inside the cone, ambiguity exists. The objects inside the cone are defined as the ambiguous candidates.

5.2.2 Gaze probe decluttering

Once a set of ambiguous candidates are determined, we want to declutter them. The novelty of this method is that we use gaze variants to declutter the ambiguous candidates. Firstly, we find the average centroid of all the candidates. Using this average position as the centre, we reposition all the candidates around it like a circle with equal intervals. As illustrated in Figure 5.3b, the gaze probe is a circle centred with the average position but not the centre of the gaze cone projection. It is because gaze lacks the depth dimension which is required to set the new positions of the candidates. Also, gaze

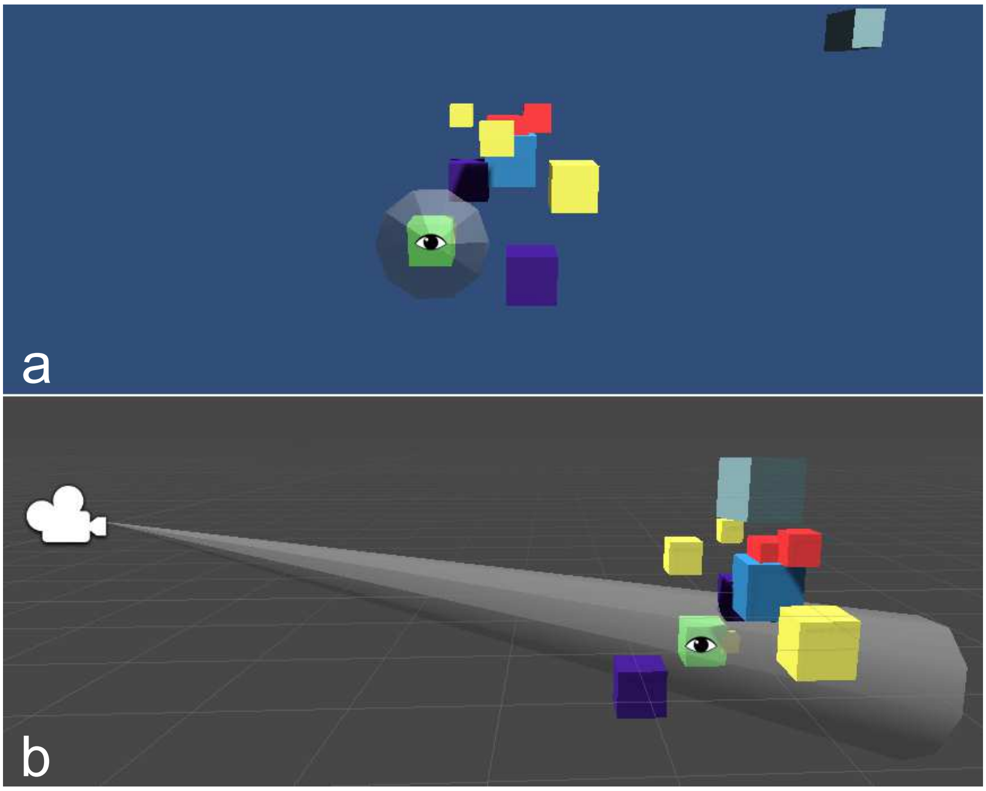


Figure 5.2: *Illustration of a gaze cone example. (a)The scene that the user sees. The user is looking at the green cube. The top of the cone is determined by transforming the 2D gaze position on the screen to the camera's near clip plane. (b)The right view of the scene. The gaze cone shoots from the camera's near clip plane in the direction to the gaze. In this frame, there are two selection candidates, the green cube and the small yellow cube behind it.*

keeps jittering, but the candidates that the gaze cone covers do not change too much with the jittering. This can filter the gaze input and stabilise the new positions of the ambiguous candidates.

Because the gaze cone projection is a circle, it is designed that the candidates declutter in a circular pattern for visual consistency. In BalloonProbe (Elmqvist and Tudoreanu 2007), the objects are projected onto the sphere surface as their new positions, so the candidates are with different depths. We instead separate the objects into their new positions with the same depth which is determined by the average centre so that they are scattered on the same vertical plane. This is to avoid new occlusions after the decluttering. Each candidate is repositioned to its closest spot on the circle (see Figure 5.3a as an example).

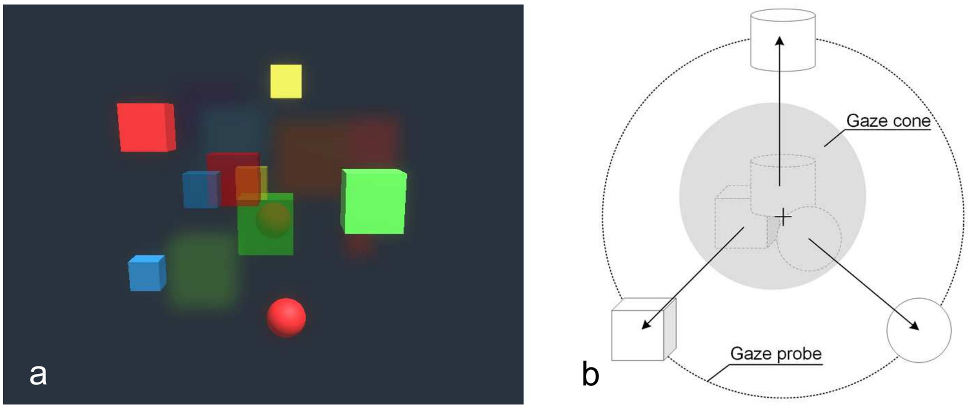


Figure 5.3: *Gaze probe. (a) An example of a scene when a gaze probe is applied. The semi-transparent cubes in the middle are the original locations of the ambiguous candidates. Note they are not displayed in real applications because of distraction. (b) Overview of the gaze probe. The grey disk is the projection of the gaze cone and the central crosshair indicates the average centroid of the three candidates that are covered by the grey disk. The cluttered objects are separated equally around the circular gaze probe which is centred with the crosshair. The centre of the gaze cone projection is not necessarily aligned with the crosshair.*

A mask for blurring out the other objects is used in order to highlight the ambiguous candidates and the users can only select from the outstanding objects (Figure 5.3a). The background is gradually blurred out and in the meantime, the objects are animated from their original positions to the new positions. To avoide the Midas Touch problem, the mask is triggered by a combination status of the gaze and gesture. When it detects an eye fixation over $200ms$ and the hand is swiping towards the gaze position, the ambiguous candidates are determined and they will start to separate in the circular pattern. Once the selection is aborted or the target is locked and confirmed, the mask disappears and the distracting candidates recover their original positions. Whether the selected target recovers its position depends on the purpose of the tasks. For visualisation purposes only, it may recover its position with a highlighted visual effect. For manipulation purposes, it may stay at the new position and moves with the hand.

The fixation threshold is set to $200ms$ because a delay over $250ms$ can be clearly perceived and the users may start to feel the system is lagging (Kangas et al. 2014b). Less than $200ms$ may make the system too sensitive to prevent unexpected triggering.

The radius of the gaze probe cannot be too small as it is difficult to accommodate the candidates with clear gaps. A clear gap should satisfy the condition that no ambiguity will be detected by the gaze cone in the new positional layout.The radius cannot be too large either because it will cost users more time to relocate the final target for the fine selection, either manually or by gaze. However, we prefer to use gaze selection here because gaze is much faster in target acquisition when there are no proximity or occlusion problems.

5.2.3 State transition

In a typical drag-and-drop task that involves selection and manipulation, this two-step selection for disambiguation can be easily integrated. Figure 5.4 gives the state transition graph to present how this is done.

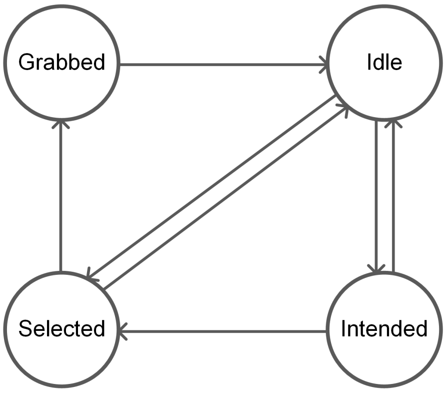


Figure 5.4: *The state transition graph for selection and manipulation tasks that may involve disambiguation using the proposed technique.*

There are four states: *Idle* state indicates that the system is in standby, no interaction is taking place. *Intended* state indicates that ambiguity is detected, so the ambiguous candidates are presented. Further fine selection is needed. *Selected* state indicates that one target is selected. No ambiguity exists at this stage. It only shows a selected status, so it cannot be manipulated yet. *Grabbed* state indicates that the selected target can be manipulated by the user.

The path *Idle* → *Selected* → *Grabbed* is for the non-ambiguous scenario, and *Idle* → *Intended* → *Selected* → *Grabbed* is for ambiguous scenarios. The interactive patterns for the users are the same in both paths for consistency purposes. When idle, the user finds the region of interest (ROI) by looking and makes a swipe gesture toward the ROI. If only one object is detected, this object will be registered as "selected" directly, and the user can grab and manipulate it without constantly looking. If multiple objects are detected, the user needs to further refine the selection before grabbing. To keep the interactive pattern consistent with the previous situation, we allow the user to refine the selection using merely the gaze cone without any other manual gestures. It is natural for the user to look at the final target when the candidates have scattered apart, which does not even cost extra cognitive effort to notice the additional gaze selection. In other words, the only difference between the two paths is an inconspicuous saccade. In both scenarios, the object marked as "selected" will be visually highlighted. *Selected* → *Idle* and *Intended* → *Idle* can be considered as deselection. *Grabbed* → *Idle* indicates dropping the target when the manipulation is finished.

Table 5.1 describes the triggers of the state transition. The gaze, gesture and candidate columns represent the components of the triggers, they need to be satisfied at the same time to trigger the state transition. A short dash means there is no specific constraint of this condition.

Gaze indicates a duration of a fixation or the existence of a gaze on the target. Note that to transit from *Selected* to *Grabbed*, only if the previous state was *Intended*, i.e., in the ambiguous scenario, a gaze on the target is required. In other words, when ambiguity occurs, the users should be able to select and deselect among the candidates, this is another reason why we use gaze-only selection here for quickly switching the target.

The gesture condition is explained using descriptive terms, in which *swipe towards* and *swipe away* indicate swiping towards or away from the gaze location, *close palm* and *open palm* are the grabbing and releasing gestures. A cursor represents the hand position is given as a positional reference when swiping towards or away from the ROI.

Table 5.1: *Triggers of the State Transition.*

State transition	Gaze	Gesture	Candidate
Idle → *Selected*	$> 200ms$	swipe towards	$= 1$
Idle → *Intended*	$> 200ms$	swipe towards	> 1
Intended → *Selected*	✓	-	$= 1$
Selected → *Grabbed*	✓ / -	close palm	$= 1$
Grabbed → *Idle*	-	open palm	-
Selected → *Idle*	-	swipe away	-
Intended → *Idle*	-	swipe away	-

The candidate condition indicates the number of ambiguous candidates that are detected inside the gaze cone. When the candidate condition is *=1*, no ambiguity exists; when the candidate condition is >*1*, ambiguity kicks in.

5.3 Experiment

The aim of the experiment is to evaluate the efficiency and accuracy of the gaze modulated disambiguation technique in selection tasks using eye tracking and gesture control, especially when the proximity or occlusion problem occurs. Therefore, a task is designed to select the only sphere from many distracting cubes. There are the following hypotheses based on the aim: comparing with the default Gaze+gesture technique, 1) the proposed technique has equivalent accuracy when no proximity or occlusion occurs; 2) the proposed technique can improve accuracy when proximity and occlusion occurs; 3) introducing disambiguation may degrade interactive efficiency.

To better understand the proposed technique, not only compared it with the default Gaze+gesture technique, I also compared it with mouse interaction both with and without the decluttering process. Thus, there are four techniques compared under three conditions, no occlusion, partial occlusion (proximity), and full occlusion (see Figure 5.6 for the examples). The four techniques are:

Mouse (M) The cubes can be dragged by the mouse. The first click on the sphere indicates a successful selection. When an occlusion occurs, he/she can drag the distracting cubes away to reveal the sphere.

Mouse+Declutter (MD) A similar probe decluttering technique using a mouse was integrated but the ambiguity was detected by ray-casting instead of gaze cone. Thus, the probe circle is centred with the mouse cursor. Each click will return how many objects are penetrated by a ray shooting from the clicking point. If it returns a number greater than one, and the first object is not the sphere, it detects an occurrence of ambiguity. A mask will appear with the ambiguous candidates scattered and highlighted as in Figure 5.3. The participant needs a second click to select the sphere if it is among the ambiguous candidates or he/she needs to right click to cancel the mask and every object goes back to their original places. Again, the first click on the sphere indicates a successful selection.

Gaze+Gesture (GG) This is the default Gaze+gesture technique. To select an object, the user needs to look at it and make a gesture, for example, a grab. If the target is occluded, the user needs to move the cubes away to reveal the sphere. Once the sphere is selected, the selection is marked as successful.

Gaze+Gesture+Disambiguation (GGD) This technique was elaborated in section 5.2. One thing to add is that in the *Grabbed* state, if the selected object is a cube, the participant can freely move it or open his/her palm to release/deselect it; if the object is a sphere, a successful selection will be admitted.

5.3.1 Apparatus

The experiment was set up in a similar but improved way as in the previous chapter. Participants sat $55cm$ away from a desktop screen running the experiment built with the Unity3D game engine. The display was a Lenovo LS2323 23” wide LCD monitor with a frame rate set to $60Hz$. The resolution was 1920×1080. A Tobii EyeX tracker was mounted on the bottom edge of the display with estimated 0.4 degrees of visual angle accuracy and the sampling rate used was the same as the frame rate. The viewing was binocular and the calibration was conducted with both eyes. The hand was tracked by a Leap Motion sensor placed facing up about $50cm$ away from the display and $17cm$ lower than the eye tracker to prevent infrared light interference as we discussed in the previous chapter. The SDK provided by Leap Motion was used for gesture recognition. The mouse was a Logitech M280 with the sensitivity set at 1000 DPI (dots per inch).

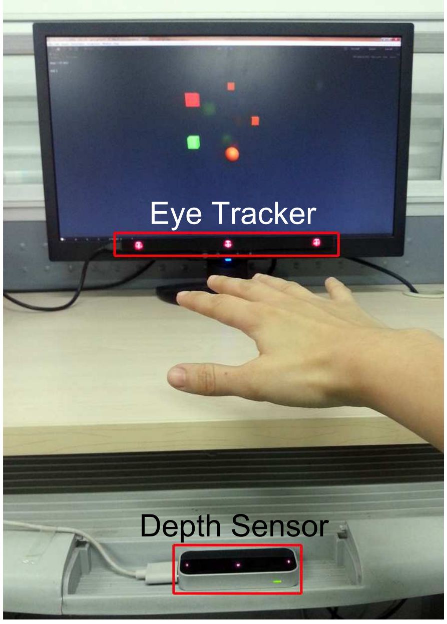


Figure 5.5: *Experiment setup.*

5.3.2 Participants

Twelve volunteers (two females) participated in the study, aged 22 to 30 (Mean $\pm$ SD = 24.9 ± 2.3). None of the participants had any eye movement, hand movement or neurological abnormalities. The participants either had adequate natural visual acuity or corrected vision with glasses. All participants reported being right-handed. Written consent has been obtained from each participant after explanation of the experiment. Before starting the tasks, participants were asked to answer some background questions by rating a $5-$point Likert scale from $1-$ Strongly disagree to $5-$ Strongly agree. All the participants stated that they mainly use mouse and keyboard for computer interaction. Most participants never used mid-air gesture control except for four participants (Mean ± SD = 1.5 ± 0.8). As for using eye tracker as an interaction interface with computers, only one reported he had some experiences (Mean ± SD = 1.2 ± 0.6).

5.3.3 Procedure

The user study started with a brief introduction and a demographic questionnaire as shown in Figure B.1. The participants were instructed to sit fairly still without restricting their head movements. Before recording the experiment data, a 7-point calibration was performed (three points separated equally on the top edge and the bottom edge respectively, and one point in the middle). Then one technique at a time was described to the participants and they were asked to practice the technique until he/she felt confident. The practice usually took less than 5 minutes for each technique. After that, the software started to record the data of this technique. The order of the four techniques was randomised. Each technique had 90 trials tested, in which each occlusion condition was tested in 30 trials. The order of the 90 trials was randomised to guarantee that the occurrences of the three different occlusion conditions were randomised as well. Thus, 12 participants × 30 trials × 3 conditions × 4 techniques = 4320 runs were tested in total.

The scene was the same for each trial which contained one sphere and ten cubes spreading within $10°$ of visual angle in the middle of the full screen. The task was to select the only sphere amongst the ten cubes. There was another cube at the top right corner of the scene marking the destination of where to drag the sphere to. The destination was about $10°$ of visual angle away from the centre of the clutter of cubes. The size, colour and position of the sphere and the cubes were randomly generated at the beginning of each trial. Once a successful selection was admitted, i.e., the sphere was selected, it would be automatically moved to the destination and disappear when it collided with the object. The automatic movement after the selection was only for indicating a successful trial, it was not included in the data recording because our aim was to evaluate the selection performance, not including the manipulation following it.

As selection is more about position acquisition, the orientation of the objects was eliminated from the task implementation in order to remove redundant noises. Thus, the gesture could only employ 3-DoF on the virtual objects.

After all trials were completed, the participant was given a SUS (System Usability Scale) (Brooke 1996b) questionnaire to complete for evaluation of each technique (see Figure B.2). A post-task interview was also conducted to collect qualitative feedback.

The whole process typically took 50 minutes to complete for each participant. The experimenter would remind the participants to have a break every 15 trials but they could skip the break and continue with the trials. They could still ask for a break at any time during the tests when they felt necessary.

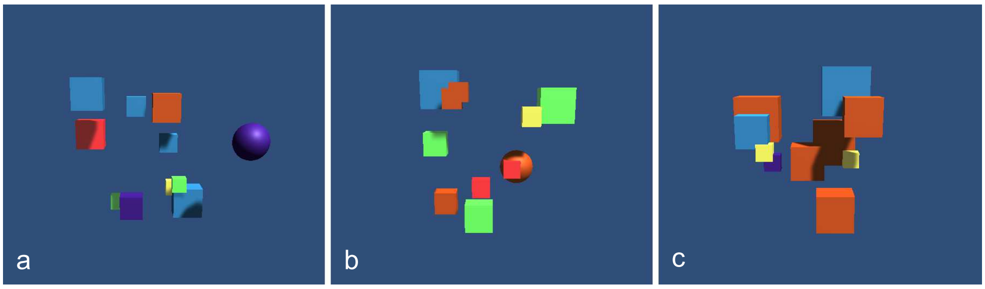


Figure 5.6: *Examples of the trial scene under three conditions. (a) No occlusion. (b) Partial occlusion (proximity). (c) Full occlusion.*

5.3.4 Measures

The quantitative evaluation included three parts: the task completion time, the error count, and the SUS score. The qualitative evaluation included a post-task interview asking for feedback on the overall experience and what the user liked and disliked of each tested technique.

Task completion time The timer started when the scene was displayed and stopped when the sphere was grabbed and marked as "selected" when it was about to automatically move to the destination.

Error count Two types of errors were measured in the experiment, the *cube error* and the *decluttering error*. The cube error count (N_{ce}) increases by one when a cube is selected instead of the target sphere. The decluttering error count (N_{de}) increases by one when no successful selection is registered in a decluttered scene, i.e., a scene that the ambiguous candidates are presented in a circularly scattered way (Figure 5.3a). Typically, based on the technique design, the cube error will hardly occur in mouse+declutter and Gaze+gesture+disambiguation, while the decluttering error will hardly occur in mouse and Gaze+gesture. The total error count of one trials is $N_{ce} + N_{de}$.

SUS score The same SUS questionnaire as in the previous chapter was presented with 10 statements with a $5-$point Likert scale from $1-$ Strongly disagree to $5-$ Strongly agree. The range of a SUS score is between 0 and 100 from low to high satisfactory. Please refer to Section 4.5.4 for how the SUS score is calculated.

5.4 Results

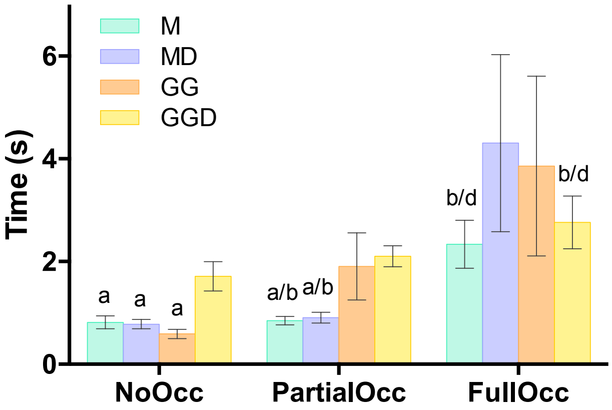
To understand the usability of the disambiguation technique, I evaluated its efficiency by measuring the task completion time, and evaluated the accuracy by measuring the error count. The SUS score and interview feedback revealed user preference among the different techniques.

5.4.1 Usability

The usability of the four techniques was evaluated under three occlusion conditions, thus twelve combinations of $technique \times occlusion$ were tested, for each combination data of 30 trials was collected from each participant. I obtained the average of the 30 trials from each participant and applied a repeated measures two-way ANOVA to estimate the impact of the task completion time and error count introduced by the different techniques and occlusion conditions. A *post hoc* Tukey test was applied to identify specific techniques and occlusion conditions who caused the significant differences. All statistical significance were determined at the level of 5%.

Efficiency The variance analysis showed that the different techniques were associated with different completion times, $F(3, 33) = 10.60$, $p < .0001$. The different occlusion conditions affected the completion time as well ($F(2, 22) = 81.32$, $p < .0001$). The result also yielded a statistical significance of the interaction between the two factors ($F(6, 66) = 18.75$, $p < .0001$). Thus, the task completion time depends on the technique and also the occlusion level.

It was more interesting to look into the impact of the techniques, so I further analysed which technique was significantly different to the others under each condition. Figure 5.7 illustrates the task completion time for each technique grouped by the occlusion conditions.



a: significantly different with GGD, $p < 10^{-2}$
b: significantly different with GG, $p < 10^{-3}$
d: significantly different with MD, $p < 10^{-4}$

Figure 5.7: *Completion time for each technique under the three occlusion conditions. Error bar indicates the standard deviation.*

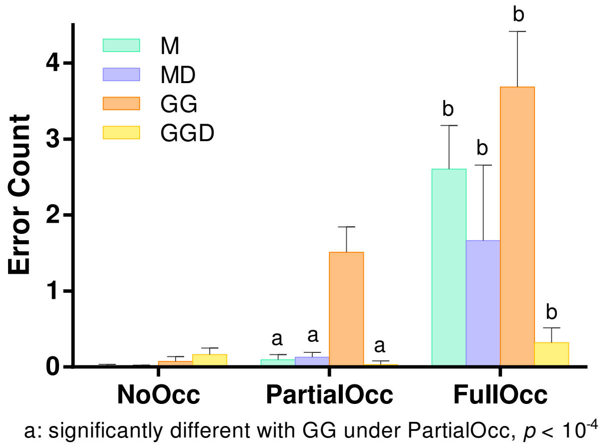
The *Post hoc* results can be simply interpreted this way. When there was no occlusion, Gaze+gesture with disambiguation took significantly longer time than the other three; when there was partial occlusion, the two techniques using mouse were more efficient than the two using Gaze+gesture no matter whether there was disambiguation; when there was full occlusion, the four techniques all differ with each other, and the efficiency sorted from high to low is $M > GGD > GG > MD$. This result showed no advantage in efficiency of the disambiguation technique when occlusion was not severe. However, it did improve the selection efficiency to a close level of the mouse when full occlusion happened.

It was not our main concern about the impacts of the occlusion condition because it was assumed that increasing the level of occlusion would lead to longer completion time in all techniques. I still briefly ran the Tukey test to test this assumption. The result showed that only full occlusion impacted the efficiency of the two techniques using a mouse. The two techniques using Gaze+gesture followed our assumption, only that the increasing rate of completion time was much larger when there was no disambiguation. Comparing to the dramatic increase of Gaze+gesture, adding the disambiguation yielded a fairly flat increase. It indicates that Gaze+gesture was extremely sensitive with occlusions and the disambiguation technique could largely alleviate this sensitivity.

Accuracy The variance analysis result shows that the number of errors occurred was associated with the level of occlusion ($F(2, 22) = 128.2, p < .0001$) and interaction technique ($F(3, 33) = 152.9, p < .0001$). The interaction of this two factors also yielded significance, so the relationship between the number of errors and the techniques is affected by the level of occlusion ($F(6, 66) = 86.28p < .0001$).

Figure 5.8 shows the average number of errors occurred in a trial of each technique under each occlusion condition. Note that this result combined the cube errors and the decluttering errors. For a breakdown of the two errors in each bar please refer to Table 5.2. The *Post hoc* test reveals that when there was no occlusion, the occurrences of both errors were close to zero in all four techniques; when there was partial occlusion, only Gaze+gesture had 1.51 cube errors while other techniques barely had any errors; when there was full occlusion, only Gaze+gesture featured with disambiguation had nearly zero occurrences of errors and the sorted accuracy among the four techniques is $GGD > MD > M > GG$.

A comparison across the three different occlusion levels grouped by techniques shows that Gaze+gesture was prone to errors as long as there existed occlusion. The two mouse techniques were capable of fine selection, so they could still perform well in the partial occlusion condition. However, in the full occlusion condition, the cube error count of the mouse technique would greatly surge. Adding decluttering in the mouse technique helped reduce the cube error but increased decluttering error. The overall error count of the mouse with decluttering was still less than the mouse technique. The fluctuation of the accuracy maintained fairly flat across all occlusion levels for Gaze+gesture+disambiguation. It suggests that this technique is robust to prevent both types of errors.



a: significantly different with GG under PartialOcc, $p < 10^{-4}$
b: significantly different with all others under FullOcc, $p < 10^{-4}$

Figure 5.8: *Error count of each technique under each occlusion condition. Error bar indicates the standard deviation.*

Table 5.2: *Breakdown of the average error count per trial.* $\bar{N}_{ce}$ *average cube error count,* $\bar{N}_{de}$ *average decluttering error count.*

	NoOcc		PartialOcc		FullOcc	
Technique	$\bar{N}_{ce}$	$\bar{N}_{de}$	$\bar{N}_{ce}$	$\bar{N}_{de}$	$\bar{N}_{ce}$	$\bar{N}_{de}$
M	.014	0	.10	0	2.61	0
MD	0	.01	0	0.13	0.01	1.66
GG	.08	0	1.51	0	3.68	0
GGD	.003	.16	.01	.03	.003	.32

5.4.2 Preference

Both quantitative and qualitative methods were applied to evaluate the user preference among the four techniques. A SUS score between 0 to 100 of each technique was obtained. This score was an overall evaluation without considering the occlusion conditions separately. The participants' responses to the interview helped us understand what they liked and disliked about each technique.

SUS Score The SUS score from high to low is mouse (84.38), Gaze+gesture +disambiguation (80.83), mouse+declutter (63.54), Gaze+gesture (61.04).

Mouse is a mature and the most pervasive interaction technique for the users, so it was supposed to score high as a reference for us to evaluate the other techniques. It could score higher if it was not tested in the full occlusion condition. The low score for mouse+declutter indicates that the decluttering technique was inappropriate for mouse interactions. However, it was suitable for Gaze+gesture as Gaze+gesture+disambiguation obtained the second highest score even though the efficiency was degraded. Compared to the low score of Gaze+gesture, it reveals that users prefer steady accurate selection instead of inaccurate fast selection.

Qualitative Feedback

- **Mouse** Every participant liked this technique because it was straightforward even though it required the user to drag the cubes one by one in the fully occluded condition. One participant pointed it out that mouse

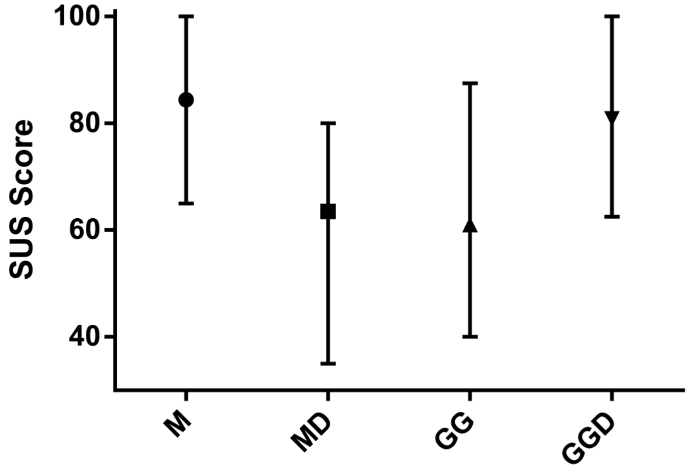


Figure 5.9: *SUS score mean of each technique. The top and bottom whiskers indicate the max and minimum score.*

would not be appropriate to be used in VR and AR compared to the gesture-based interactions.

- **Mouse+Declutter** Most participants had a neutral preference of this technique. The complaints about this technique were typically on the decluttering. One participant suggested why not enable mouse controlled rotation instead of decluttering the cubes as the previous one was more straightforward and common in 3D mouse manipulation. Several participants mentioned that the single point of the mouse cursor made it difficult to find an invisible target without prior knowledge of the scene, so it required clicking on many positions, which was not as easy as sweeping through a large area in one go.

- **Gaze+Gesture** This technique was the most popular in the no occlusion condition as it was fast and accurate. The participants had great passion to try out new interaction techniques except for one participant who preferred well-developed traditional techniques. No exhaustion of eyes was reported. However, hand tiredness was reported as in the occluded condition, they had to move every blocking cube away one by one, including those who were grabbed by mistake. It could be noticed that when the participants could not grab the object they expected, they tended to try more times and gradually grab with more strength in a tight fist. This accelerated their arm fatigue.

- **Gaze+Gesture+Disambiguation** No eye fatigue or arm tiredness was reported in this technique. Compared to the Gaze+gesture technique, the participants commented that this technique was more relaxed as it typically only required one grab in each trial. They were not aware of the gaze cone design but they preferred this technique to the mouse+ declutter technique. One participant commented that this technique felt like it was designed following the naturalness of human gaze, as it worked with clusters instead of pixels.

  One participant particularly reported that the decluttering was enjoyable in interaction. However, some found the trigger to declutter the ambiguous objects was confusing because of the cursor design. The participants were instructed to swipe towards the direction they were looking at using the cursor as a reference of their current hand position so they could decide on which direction to swipe. They tended to ignore the cursor but swipe based on their own proprioception. It could work in most cases but sometimes they could swipe to the opposite direction and got frustrated and confused why the trigger did not work. Some of them would move the cursor to the object they wanted to pick up as how they used the mouse cursor.

  Two participants mentioned that the selection was disabled during the decluttering animation, which was to animate the cluttered objects from their original positions to the new positions. They wished to be able to select once it started decluttering. There was no need to wait until the objects were in position. This is a good point as it can help improve the efficiency of this technique, and it specifically suits the Gaze+gesture technique. The same animation settings were applied in mouse+declutter but nobody complained the animation was too slow, we assume that was because gaze was much faster than manually moving the mouse on the target. This might also be part of the reason that why mouse+declutter had fewer errors but much longer completion time.

5.5 Discussion

The three hypotheses of the experiment received positive answers. It was validated that Gaze+gesture was accurate either with or without the disambiguation technique when there was no occlusion. While in occluded conditions, the disambiguation reduced the surging error count of the original Gaze+gesture down to nearly zero. However, the disambiguation was not advanced regarding efficiency, especially in the unoccluded condition.

The mouse was measured in the experiment as a benchmark and supported that the proposed technique did well in preventing errors but still not as fast as the mouse. Moreover, applying only the decluttering technique to the mouse without conical ambiguity detection failed to achieve competent accuracy as the Gaze+gesture+disambiguation technique.

As Cashion et al. (2012) commented, there was no best technique for all situations. Each technique is dependent on specific conditions to fit, or to be tailored, as the best solution.

5.5.1 Relationship between accuracy and efficiency

Overall, it was observed that the efficiency was positively correlated with the accuracy of every technique. The task completion time increased with the growth of error count from no occlusion to full occlusion. Moreover, the coefficient of the correlation between completion time and error count was different with each technique. Some had fewer errors but longer completion time, and vice versa, such as the two mouse techniques.

However, as the unoccluded condition involves no errors, the completion time under this condition was irrelevant to the number of errors. It defined the nature of each technique. The Gaze+gesture+disambiguation technique is a good example. Its performance without occlusions was already slower than the others, so its poorer efficiency was not caused by the errors but the nature of its design. This could explain why the proposed technique had better accuracy but poorer efficiency.

The technique used a coarse-to-fine two-step design in selection. Compared to the direct selection in mouse and Gaze+gesture, it would double the completion time because it basically consists of two direct selections. As shown in the fully occluded condition, the completion time of the mouse increased because of the errors while the Gaze+gesture+disambiguation barely had any error. However, their completion times were close to each other. It indicates that a balance between the accuracy and efficiency has been reached using the Gaze+gesture+disambiguation technique in the fully occluded condition. Similarly, Gaze+gesture also reached a balance with our proposed technique in the partially occluded condition but only earlier because of its sensitivity to occlusions.

This is in line with the Fitts' Law (Fitts 1954) which points out that reaching a small target will cost more time. It suggests that accuracy and efficiency cannot be fulfilled at the same time in certain circumstances. Although not quite the same with reaching a small target, we would consider reaching an occluded object shares this common feature and requires a trade-off between accuracy and efficiency in interaction design. Elmqvist and Tudoreanu (2007) reported the same argument in an occlusion management comparison study.

5.5.2 Comparisons among disambiguation techniques

I have also implemented another two disambiguation techniques in a pilot study using the transparency method and the multiple views method which are reviewed in Section 2.4.2. In the transparency method, all ambiguous candidates are set to semi-transparent to visualise the occluded objects, and then users are able to move their hand in the depth direction to reach them. In the multiple views method, the users are able to rotate the camera using specific hand gestures, thus the occluded objects can be revealed when the viewing angle changes.

I did not adopt these two methods because they are more suitable for the occlusion problem, whether they can be applicable for the proximity problem depends on the positional layout of the virtual scene. To accurately access a target using these methods, it still requires the virtual objects to be scattered sparsely because of the intrinsic lack of accuracy of both eye tracking and virtual hand techniques. Therefore, I adopt the volumetric probe method because it involves a step to relocate the ambiguous candidates.

Furthermore, the transparency method provides a weak 3D spatial perception when all the semi-transparent ambiguous candidates are piled up in the gaze direction because of their colour interference with each other. This design can be improved such as revealing the occluded objects layer by layer with the relative movement of users' hand in the depth direction, i.e., hiding the occluding layers completely without using semi-transparency.

However, both methods utilise the 3D nature of the virtual hand metaphor, and the multiple views method also takes advantage of the rich expressions of gesture control. Besides, it is natural to navigate around the scene to see what is behind so rotating the viewport is a more intuitive method, especially in VR that users can simply walk around to access the occluded virtual contents. In this sense, the volumetric probe method provides a not as natural but efficient way to enable fast acquisition of the occluded contents because it saves much more energy of adjusting a proper view.

In short, each method has its best cases to fit in. The real challenge is to choose the suitable method depending on the case. A possible approach could be context awareness which applies different techniques depending on the virtual contents in the gaze cone patch.

5.5.3 Limitations

The current design of this disambiguation technique has limitations to be used when the ambiguous candidates are of various sizes and shapes. For example, a case that a small object fully occluded by a very large object that is much bigger than the gaze cone projection can be better dealt with using transparency or changing the viewport.

Selecting an occluded object does not always involve positional manipulation, such as for display-only purposes. Besides, repositioning loses the context of the original scene, so possible confusion can be introduced when the selected ambiguous candidates share the same features, especially in shape and colour.

Moreover, there is a limitation of the maximum number of ambiguous candidates the gaze probe can accommodate as the decluttering circle has limited space. Possible solutions are to add more layers of circles or adjusting the circle size but the usability remains unknown.

5.5.4 Design recommendations

Firstly, the usage of the proposed decluttering technique should be coupled with volume selection, such as cone selection used in this chapter, especially when prior knowledge of the scene is missing. In mouse+declutter experiment, the decluttering technique did not help the mouse improve accuracy not only because of the lack of gaze cone in ambiguity detection but also because of the lack of prior knowledge of the scene. Imagine that the size and rough position of the target were known before selection, the ray-casting technique might still achieve satisfying performance. When there was nothing known about the target, the only chance to find it was to try clicking on every possible pixel especially when the target was fairly small.

Secondly, cone selection is preferred than ray selection when both occlusion and proximity error may occur, and it is not sensitive to the missing of prior knowledge of the scene. The original intention of designing the gaze cone instead of using a gaze ray was to cope with the gaze jittering and the proximity problem as a gaze ray could only reach objects occluded in depth but not with offsets on the planar plane. Thus, in the mouse+declutter technique, as the mouse was stable and free of planar offset, i.e., the proximity problem, only a ray was applied for its ambiguity detection. Clearly, the result suggests that the cone selection not only helps ambiguity detection on the planar plane but also on occlusion in depth. Furthermore, it has the advantage to be used in scenarios either with or without prior knowledge of the virtual scene.

Lastly, the Gaze+gesture technique can be applied in more VR or desktop applications with the disambiguation technique enabled, such as virtual interior design, digital LEGO games and 3D model design. These applications are 3D who contain rich information, so enabling easy selection when an occlusion occurs can be helpful for them. Figure 5.10 gives an example of using this disambiguation technique in a virtual scene of a car model design.

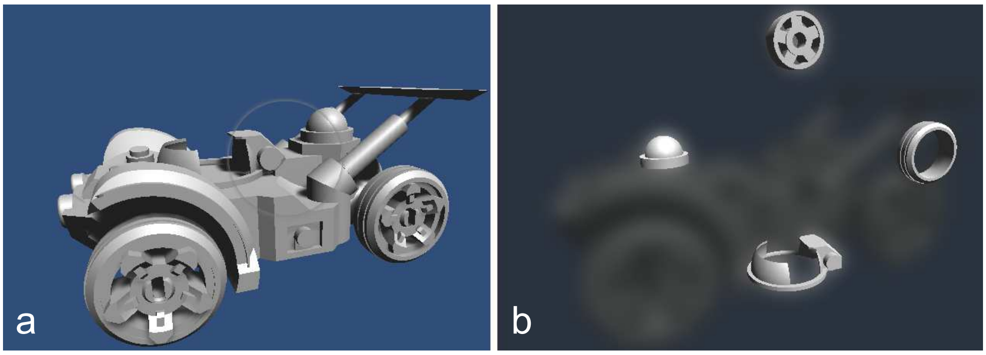


Figure 5.10: *An example using the gaze modulated disambiguation technique to display occluded car parts in a modelling application*[1]. *(a) The original model of a car. The circle indicates the gaze area which is invisible in the real application. (b) The corresponding decluttered ambiguous candidates. The wheel on the top and the tyre on the right were fully occluded before the decluttering.*

5.6 Summary

This chapter presented a two-step disambiguation technique to facilitate gestural selection in occluded 3D virtual environments, which combined gaze cone for ambiguity detection, i.e., coarse selection, and gaze probe to declutter the ambiguous objects for fine selection. The user study shows that this technique could prevent selection errors caused by inaccuracy of eye tracking and occlusion. The enhanced user preference supported the positive effect of the technique.

By comparing with a well established pointing technique, the mouse pointing, we understood that the current usability was acceptable but it could still be improved especially in efficiency, in order to be pervasively used in daily life. Moreover, the disambiguation technique was designed based on the advantages of gaze, so it was not as suitable for mice. However, gaze modulated pointing can be combined with other manual inputs instead of gesture control, such as touchscreens and keyboards, only that the naturalness and feasibility may not be as good depending on what manual input is adopted.

[1] The model was obtained from http://www.123dapp.com/123D_Design/car/4427801

Chapter 6

Conclusion and Future Work

6.1 Conclusion

6.1.1 Findings

In this thesis, I investigated the feasibility of multimodal interaction techniques that combined eye tracking and gesture control, especially in continuous manipulation tasks. This work is supported by empirical evidence and data which show that gaze modulated gesture control is capable of providing a stable and efficient user experience in natural interactions within virtual space. Three research questions are addressed, 1) what is the spatio-temporal relationship between gaze and indirect input devices (Chapter 3); 2) is there any perceptual conflict generated by the multimodal integration (Chapter 4); and 3) how to deal with ambiguities during this multimodal interaction (Chapter 5).

In answer to the first research question, I found that:

- the gaze movement and indirect input devices were linearly related;
- in most cases, gaze movement led the movement of virtual cursor/hand both temporally and spatially.

The results show that gaze lead in gaze-mouse coordination is comparable to the natural eye-hand coordination. Combining with existing studies, it suggests that the gaze lead pattern in eye-hand coordination tasks with both direct and indirect devices is comparable to the gaze lead pattern in eye-hand coordination with physical hands. In this case, we can broaden the findings of eye-hand coordination in the physical world into indirect human-computer interactions that utilise a tool or medium to interpret our physical hand movement into the virtual space.

In answer to the second research question, I found:

- a perceptual conflict, the spatial misperception problem which describes the phenomenon that user's proprioception of their hand relative to the virtual space could not maintain constant with their visual perception when the virtual cursor/hand was modulated by their gaze;
- comparing to Scaling and the original Normal method, the preferable methods, Magnet and Dual-gaze, both remained spatially consistent when handling the spatial misalignment, which suggests that a constant spatial perception better conforms to the naturalness in interaction design and benefits the user experience;
- the two preferred methods could also reduce arm fatigue to the minimum in specific scenarios;
- each method is specialised to be used in specific tasks.

It is important to identify and solve this misperception problem because it can cause unpredictable interactive interruptions during continuous manipulation such as dropping an object when moving it. Interactive interruptions indicate an unstable interaction design which provides unpleasant user experience, so it worth solving this misperception problem to stabilise the performance of the interaction technique and enhance user experience.

In answer to the third research question, I proposed a two-step coarse-to-fine technique using the gaze variance, in which a gaze cone is firstly used to detect ambiguities such as a cluster of virtual objects, and then a gaze probe is used to declutter the cluster for visualising and enabling gaze pointing to specify the target objects.

This ambiguity technique was compared with the original Gaze+gesture technique and the conventional mouse pointing, in which I found:

- when the scene is fully occluded, the proposed technique can achieve better performance than mouse pointing;
- when occlusion is not severe, the proposed disambiguation technique needs to improve its efficiency to be comparable with mouse pointing.

It is indispensable to solve this ambiguity problem because proximity and occlusion can easily occur in any virtual environments that contain rich information. In this case, fine grained selections and manipulations are basic interactive requirements. Failure to provide fine grained interactions can degrade the accuracy and efficiency of gaze modulated gesture control, especially given that accuracy is often related to serious problems such as safety. This disambiguation technique dramatically reduced error occurrence in fine selections with a reasonable sacrifice in efficiency.

Overall, answering the three research questions fills the gaps in the design of gaze+gesture technique, which helps provide a more stable and robust natural 3D interaction in VR/AR applications, such as virtual surgery and games. This thesis contributes to provide a practical example of developing multimodal interaction techniques to inform future design of natural user interfaces in a timely manner.

6.1.2 Recommendations

This research led an interdisciplinary study that combined qualitative methods and quantitative methods such as experiment design, correlation analysis of behavioural patterns, interaction technique prototyping, comparative study, survey, and statistical analysis. Throughout the research process, I have gained valuable hands-on experience, from which I want to highlight the following lessons I have learned for the benefit of the future research practice of myself and interaction designers who will conduct related studies.

- Fitts' Law (Fitts 1954) suggests that larger targets are easier to access for both accuracy and efficiency. Besides, eye tracking and gesture control are limited in pointing accuracy in spite of that disambiguation techniques are proposed for resolving this problem. Hence, it is still recommended to design the interactive components in VR/AR such as virtual objects and buttons with a decent size for the benefit of user's convenience. However, to obtain data for learning eye-hand coordination patterns either in a physical or virtual environment, it is recommended to design the experimental stimuli with a smaller size for the sake of data quality. This is because smaller stimuli require more coordination demands which can reduce the distance disparity between related eye movement and hand movement data. More coordination demands can also help reduce distractor interference. Detailed rationales are discussed in Chapter 3 and Chapter 4.

- The spatial misperception problem needs to be considered when gaze modulated pointing is combined with absolute and indirect input devices in continuous manipulation, for example, stylus-based haptic devices. The human arm reach should also be considered in gaze modulated gesture control. This is discussed in Chapter 4.

- At most one infrared light facilitated tracker is recommended when multiple trackers are adopted into the interaction. Otherwise, the trackers should be deliberately positioned to avoid light interference. This is discussed in Chapter 4.

- The design of cone selection is recommended when error tolerance is required both on the planar plane (2D) and in depth (3D). Moreover, the decluttering technique can help visualise occluded scenes without positional prior knowledge of the virtual contents. This is discussed in Chapter 5.

6.1.3 Implications

The findings of this thesis can be of interest to the general audience who are interested in 3D interaction because gaze modulated gesture control can be introduced in many applications that require fast acquisition and expressive manipulation of 3D objects to enable natural user experience.

For example, in the traditional 3D design process, the common way is to build a conceptual prototype to verify the ideas by iteratively revising the layout until expectations are met. This process involves repetitive selection and manipulation of 3D components of the scene, such as a virtual presentation of interior design and furniture arrangement, virtual LEGO demonstration (Oh and Stuerzlinger 2004), and drug design that involves 3D molecular structure modelling.

The combination of gaze and gesture configuration is particularly useful with non-desktop displays including large and remote displays, smart TV, VR cave, and VR/AR headsets. The conventional way to interact with large and remote displays is via a remote control. There are studies using touch screens to act as the remote control, which enable natural interactions by 2D touch gestures (Stellmach and Dachselt 2012, 2013). Some VR kits provide hand-held game controllers to replace the remote control, for example, HTC Vive which directly enables 3D natural interactions. It tracks the 3D positions of the game controllers and enforces actual interactions by the buttons on the controller. Buttons can provide stable and reliable control but they are less expressive than natural hand gestures. The naturalness of gaze and mid-air gestures is specifically intuitive to support wearable VR/AR headsets. For example, eye tracking and gesture control are now available for integration with Oculus Rift. Jalaliniya et al. (2015) also developed a prototype combining MAGIC pointing with head-mounted displays, which proved a possibility to integrate gaze modulated pointing in head-mounted VR displays.

In summary, this thesis has important implications for the design of multimodal interactions. On the one hand, all the aforementioned possible applications have a potential risk of the spatial misperception problem and ambiguities. This thesis helps identify these problems and provide possible solutions along with other useful design recommendations. On the other hand, the thesis provides a paradigm of designing multimodal interaction techniques by following the technical pathway that firstly understanding the natural correlated behavioural patterns of the sensory modalities to be integrated, followed by investigations of possible perceptual conflicts and interaction ambiguities that may be generated by the multimodal integration.

In the next section, further research directions derived from this thesis will be discussed.

6.2 Future Work

6.2.1 Improvement of current study

The first direction of future work is to refine the current design of gaze modulated gesture control by improving the usage of gestures in three aspects: 1) design gestures for more tasks; 2) solve possible Midas Touch problem of gestures; and 3) utilise the 3D feature of gestures.

In this thesis, only the most basic task, drag-and-drop, is used for demonstrating the idea of gaze modulated gesture control. This task only involves the two most basic interactive gestures, grab and release. For more complicated tasks, more gestures need to be designed. As we know, the human hand has a very complex kinematic structure combining multiple joints and links, which can be tracked for up to 27-DoF for the whole hand (Agur and Dalley 2009), and even more if both hands are involved. It enables rich expressions of hand gestures which can help establish richer gesture vocabulary to convey more semantic information for more manipulations and communications.

Although more gestures can achieve more powerful interactions, we need to find a balance between simplicity and power. A basic discipline for interface design is simplicity because the cognitive capacity of an individual is limited and too complicated interactions will carry too much cognitive load which may exceed the cognitive capacity of a user. An overload cognition will introduce difficulties of interaction and harm the user experience. Cognitive capacity is a definite amount of cognition an individual has. The amount of cognition assigned to process a particular task is the cognitive load which cannot exceed one's cognitive capacity. Complicated gesture design may cost too much cognitive load of users, so we need to keep it simple to avoid cognitive overload.

Besides, more complicated gestures will introduce the Midas Touch problem for gestures. Because mid-air gesture control is also camera-based, which is the same with eye tracking, the user's hand actions can be considered as always active. When more complicated gestures are introduced, various combinations of fingers will be involved, similar gestures may exist. The transition between two different gestures may be recognised as a third gesture. This is the Midas Touch (Section 2.1.2) of gesture control. Existing solutions for this problem can be roughly divided into three categories, which are bottom-up visual cues reasoning, top-down semantic constraints, and the methods combining both (Wu and Wang 2016). Therefore, it is my intention to not only aggregate more useful gestural command into the gaze modulated gesture control but also to address the gesture Midas Touch problem it may introduce especially in continuous manipulation.

Another improvement is to refine the design of gestures to properly integrate the 3D accessibility of the virtual hand metaphor (Section 2.2.1). In the current disambiguation technique, the decluttering is implemented by scattering the ambiguous candidates to the same depth, i.e., a plane parallel to the clipping plane of the virtual camera, or the monitor. As we have discussed, the depth information is removed in this technique. Thus, to preserve the depth information, a future step is to take advantage of the 3D feature of gesture control and integrate it into the disambiguation technique, for example, revealing or hiding occluded layers following hands movement on the depth direction. Here the virtual hand works as a depth cursor for determining the interactive depth as how the depth ray and lock ray works (Grossman and Balakrishnan 2006).

6.2.2 Intelligent user interface

The second direction of future work is to enable intelligence of gaze modulated gesture control. One consideration is to integrate context awareness (Dey et al. 2001) into the interaction interface design. As I have discussed in the comparative studies of previous chapters, there is no generic technique that can be the best solution for all tasks and scenarios. For example, among the four interaction techniques discussed in Chapter 4, although they can generally be alternatives to each other, it is still recommended to use them dedicatedly to support specific tasks based on their speciality. Furthermore, the disambiguation technique described in Chapter 5 is limited to declutter objects with similar sizes and shapes. Other solutions such as transparency or multiple views should be integrated when dealing with ambiguity caused by objects with significant size and shape differences. A promising method to facilitate a more robust and pervasive solution is context awareness which applies different techniques depending on the task, virtual contents and user habits. Eye tracking can be applied for filtering out the irrelevant context because of its selective feature.

Another consideration is to make use of the behavioural patterns of gaze and hand movement, as well as their spatio-temporal relationship for prediction and modelling in user interface design.

Gaze leading features can help improve the performance of visually stimulated applications in which gaze can deliver a reliable prediction of manipulation area. The prediction provides the possibility of pre-computation for real-time display of complex graphics. A task with high coordination demands would benefit more according to this study because high coordination demanding tasks allow more lead time. Meanwhile, it needs to be considered in interface design that the gaze lead may not always be effective because hand may lead gaze in low coordination demanding tasks.

The observed backshoot in the saccade data also provides potential directional information for saccadic movement prediction. Evidence shows that microsaccade direction can reveal the direction of covert attentional shifts by moving away from the visual cue (Engbert and Kliegl 2003). This mechanism may explain the observed backshoot oculomotor behaviour (see Figure 3.6(c)), but there is somewhat of a dispute regarding this in the literature (Tse et al. 2004).

In addition, the strong correlation features between gaze and hand movements can be extracted from the collected data for visuomotor behaviour modelling, which will, for example, benefit the robotic implementation of trajectory-based tasks. The eye-hand correlation also forms unique patterns that distinguish it from gaze and hand movements that are irrelevant to each other. This difference has the potential to be applied in attention decoding for telling whether the user is focusing on the task or not. Such techniques can be used in a wide range of applications such as adaptive virtual reality, smart mobile devices, and intelligent web applications.

During the implementation of the methods discussed in Chapter 4, I observed that participants showed preferences to certain interpolation speed as they reported them as “smooth” while some others were reported as “cumbersome”. This intrigues my interest in the correlation between the variation of the interpolation speed and the variation of user’s satisfaction. This could be related to the temporal leading of gaze in eye-hand coordination but further experiments are necessary to testify this assumption.

6.2.3 Integration with other modalities

The third direction of future work is to integrate more modalities into the design of gaze modulated gesture control, such as audio and force feedback.

Interactive audio can be speech or non-speech. It is proved that in a virtual environment, the non-speech sound generated by physical modelling and auditory synthesis techniques can highly improve fidelity (Avanzini and

Crosato 2006). Speech, on the other hand, can not only deliver semantic information but also carry emotional information. Facial expression and body movement (Kleinsmith and Bianchi-Berthouze 2013) are typically used for emotional recognition, which are combined with vocal expressions for multimodal affective computing (Castellano et al. 2008). In recent years eye tracking starts to be used for emotional assessment (Alghowinem et al. 2014). Considering eye tracking benefits social interaction among multiple users in a virtual environment, or interactions with a virtual agent (Ruhland et al. 2015), there exist many interesting applications for affective recognition.

Force feedback can also be integrated into the current design of gaze modulated gesture control, for example, to augment the border perception using a vibration prompt. It can also be applied to the grab/release gestures for indicating successful gesture commands and thus improve naturalness and comfortableness. To enable force feedback with mid-air gesture control, we can only use wearable haptic devices such as gloves with actuators attached to fingertips and hand-held controllers. In recent research, ultrasound has been used for providing mid-air force feedbacks (Sodhi et al. 2013; Long et al. 2014; Sand et al. 2015). The combination of mid-air gesture control and wearable haptic devices already exist (Scheggi et al. 2015). However, mid-air force feedback provided by ultrasound devices has not been integrated with gesture control yet, which can introduce new research challenges.

As the new technologies are becoming more ubiquitous, interaction techniques and interfaces are transiting from traditional mouse and keyboard to be multimodal and natural. This thesis provides empirical insights for the design of multimodal interactions and natural user interfaces which we believe will play an important role in the next generation of human-computer interaction. And we can foresee studies in this area will flourish.

References

Accot, J. and Zhai, S., 1997. Beyond fitts' law: models for trajectory-based hci tasks. *In: Proceedings of the ACM SIGCHI Conference on Human factors in computing systems*. Atlanta, Georgia: ACM. 295–302.

Agrawala, M., Zorin, D. and Munzner, T., 2000. Artistic multiprojection rendering. *In: Proceedings of the Eurographics Workshop on Rendering Techniques*. Springer Vienna. 125–136.

Agur, A. M. and Dalley, A. F., 2009. *Grant's atlas of anatomy*. Lippincott Williams & Wilkins, 10th edition.

Alam, S. and Jianu, R., 2016. Analyzing eye-tracking information in visualization and data space: from where on the screen to what on the screen. *IEEE Transactions on Visualization and Computer Graphics*, PP (99), 1–13.

Alghowinem, S., AlShehri, M., Goecke, R. and Wagner, M., 2014. Exploring eye activity as an indication of emotional states using an eye-tracking sensor. *In: Intelligent systems for science and information*, Springer, 261–276.

Alt, F., Schneegass, S., Auda, J., Rzayev, R. and Broy, N., 2014. Using eye-tracking to support interaction with layered 3d interfaces on stereoscopic displays. *In: Proceedings of the 19th international conference on Intelligent User Interfaces*. Haifa, Israel: ACM. 267–272.

Andujar, C., Argelaguet, F. and Trueba, R., 2010. Hand-based disocclusion for the world-in-miniature metaphor. *PRESENCE: Teleoperators and Virtual Environments*, 19 (6), 499–512.

Argelaguet, F. and Andujar, C., 2009. Efficient 3d pointing selection in cluttered virtual environments. *Computer Graphics and Applications, IEEE*, 29 (6), 34–43.

Argelaguet, F. and Andujar, C., 2013. A survey of 3d object selection techniques for virtual environments. *Computers & Graphics*, 37 (3), 121–136.

Argelaguet, F., Andujar, C. and Trueba, R., 2008. Overcoming eye-hand visibility mismatch in 3d pointing selection. *In: Proceedings of the 2008 ACM symposium on Virtual reality software and technology*. Bordeaux, France: ACM. 43–46.

Ariff, G., Donchin, O., Nanayakkara, T. and Shadmehr, R., 2002. A real-time state predictor in motor control: study of saccadic eye movements during unseen reaching movements. *The Journal of Neuroscience*, 22 (17), 7721–7729.

Ashmore, M., Duchowski, A. T. and Shoemaker, G., 2005. Efficient eye pointing with a fisheye lens. *In: Proceedings of Graphics Interface 2005*. Victoria, British Columbia, 203–210.

Avanzini, F. and Crosato, P., 2006. Haptic-auditory rendering and perception of contact stiffness. *In: International Workshop on Haptic and Audio Interaction Design*. Springer. 24–35.

Ballagas, R., Borchers, J., Rohs, M. and Sheridan, J. G., 2006. The smart phone: a ubiquitous input device. *IEEE Pervasive Computing*, 5 (1), 70–77.

Behan, M. and Wilson, M., 2008. State anxiety and visual attention: The role of the quiet eye period in aiming to a far target. *Journal of Sports Sciences*, 26 (2), 207–215.

Bieg, H.-J., Chuang, L. L., Fleming, R. W., Reiterer, H. and Blthoff, H. H., 2010. Eye and pointer coordination in search and selection tasks. *In: Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications*. Austin, Texas: ACM. 89–92.

Biguer, B., Jeannerod, M. and Prablanc, C., 1982. The coordination of eye, head, and arm movements during reaching at a single visual target. *Experimental Brain Research*, 46 (2), 301–304.

Binsted, G., Chua, R., Helsen, W. and Elliott, D., 2001. Eyehand coordination in goal-directed aiming. *Human Movement Science*, 20 (45), 563–585.

Blake, J., 2010. *NUIs reuse existing skills (updated NUI definition)* [online]. Available from: http://nui.joshland.org/2010/04/nuis-reuse-existing-skills.html [Accessed 6 December 2017].

Blake, J., 2011. *Natural User Interfaces in .NET: WPF 4, Surface 2, and Kinect*. Manning Publications Company.

Blattner, M. M. and Glinert, E. P., 1996. Multimodal integration. *IEEE MultiMedia*, 3 (4), 14–24.

Bowman, D. A. and Hodges, L. F., 1997. An evaluation of techniques for grabbing and manipulating remote objects in immersive virtual environments. *In: Proceedings of the 1997 symposium on Interactive 3D graphics*. Providence, Rhode Island: ACM. 35–38.

Bowman, M. C., Johannson, R. S. and Flanagan, J. R., 2009. Eyehand coordination in a sequential target contact task. *Experimental Brain Research*, 195 (2), 273–283.

Brooke, J., 1996a. Sus-a quick and dirty usability scale. *Usability evaluation in industry*, 189 (194), 4–7.

Brooke, J., 1996b. Sus-a quick and dirty usability scale. *Usability evaluation in industry*, 189 (194), 4–7.

Bryson, S., 1996. Virtual reality in scientific visualization. *Commun. ACM*, 39 (5), 62–71.

Burelli, P. and Yannakakis, G. N., 2011. Towards adaptive virtual camera control in computer games. *In: Smart Graphics*. Springer. 25–36.

Buxton, W., 1990. A three-state model of graphical input. *In: Proceedings of the IFIP TC13 Third Interational Conference on Human-Computer Inter-*

action. Cambridge, UK: North-Holland Publishing Co. volume 90, 449–456.

Campbell, G. and Geller, S., 1980. Balanced latin squares. *Purdue University Department of Statistics Mimeoseries*, 80 (26), 3–1.

Cashion, J., Wingrave, C. and J. J. LaViola, J., 2012. Dense and dynamic 3d selection for game-based virtual environments. *IEEE Transactions on Visualization and Computer Graphics*, 18 (4), 634–642.

Casiez, G., Vogel, D., Balakrishnan, R. and Cockburn, A., 2008. The impact of control-display gain on user performance in pointing tasks. *HumanComputer Interaction*, 23 (3), 215–250.

Castellano, G., Kessous, L. and Caridakis, G., 2008. Emotion recognition through multiple modalities: face, body gesture, speech. *In: Affect and emotion in human-computer interaction*, Springer, 92–103.

Charness, N., Holley, P., Feddon, J. and Jastrzembski, T., 2004. Light pen use and practice minimize age and hand performance differences in pointing tasks. *Human Factors*, 46 (3), 373–384.

Chatterjee, I., Xiao, R. and Harrison, C., 2015. Gaze+gesture: Expressive, precise and targeted free-space interactions. *In: Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*. Seattle, Washington: ACM. 131–138.

Chen, M. C., Anderson, J. R. and Sohn, M. H., 2001. What can a mouse cursor tell us more?: correlation of eye/mouse movements on web browsing. *In: CHI '01 Extended Abstracts on Human Factors in Computing Systems*. Seattle, Washington: ACM. 281–282.

Chittaro, L. and Scagnetto, I., 2001. Is semitransparency useful for navigating virtual environments? *In: Proceedings of the ACM Symposium on Virtual Reality Software and Technology*. Baniff, Alberta, Canada, VRST '01, 159–166.

Coen-Cagli, R., Coraggio, P., Napoletano, P., Schwartz, O., Ferraro, M. and

Boccignone, G., 2009. Visuomotor characterization of eye movements in a drawing task. *Vision research*, 49 (8), 810–818.

Coffin, C. and Hollerer, T., 2006. Interactive perspective cut-away views for general 3d scenes. *In: 3D User Interfaces (3DUI'06)*. 25–28.

Crawford, J. D., Medendorp, W. P. and Marotta, J. J., 2004. Spatial transformations for eye-hand coordination. *Journal of Neurophysiology*, 92, 10–19.

De Haan, G., Koutek, M. and Post, F. H., 2005. Intenselect: Using dynamic object rating for assisting 3d object selection. *In: IPT/EGVE*. Citeseer. 201–209.

de Xivry, J.-J. O. and Lefevre, P., 2007. Saccades and pursuit: two outcomes of a single sensorimotor process. *The Journal of physiology*, 584 (1), 11–23.

Deng, S., Chang, J., Hu, S. and Zhang, J. J., 2017a. Gaze modulated disambiguation technique for gesture control in 3d virtual objects selection. *In: The 3rd IEEE International Conference on Cybernetics*. Exeter, UK: IEEE.

Deng, S., Chang, J., Kirkby, J. A. and Zhang, J. J., 2016. Gazemouse coordinated movements and dependency with coordination demands in tracing. *Behaviour & Information Technology*, 35 (8), 665–679.

Deng, S., Chang, J. and Zhang, J. J., 2013. A survey of haptics in serious gaming. *In:* De Gloria, A. (Ed.), *Games and Learning Alliance: Second International Conference, GALA 2013, Revised Selected Papers*. Paris, France: Springer International Publishing. 130–144.

Deng, S., Jiang, N., Chang, J., Guo, S. and Zhang, J. J., 2017b. Understanding the impact of multimodal interaction using gaze informed mid-air gesture control in 3d virtual objects manipulation. *International Journal of Human-Computer Studies*, 105, 68 – 80.

Deng, S., Kirkby, J. A., Chang, J. and Zhang, J. J., 2014. Multimodality with eye tracking and haptics: A new horizon for serious games? *International Journal of Serious Games*, 1 (4), 17–34.

Deubel, H. and Bridgeman, B., 1995. Fourth purkinje image signals reveal eye-lens deviations and retinal image distortions during saccades. *Vision research*, 35 (4), 529–538.

Dey, A. K., Abowd, G. D. and Salber, D., 2001. A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications. *Human-computer interaction*, 16 (2), 97–166.

Diepstraten, J., Weiskopf, D. and Ertl, T., 2002. Transparency in interactive technical illustrations. *Computer Graphics Forum*, 21 (3), 317–325.

Drewes, H. and Schmidt, A., 2009. The magic touch: Combining magic-pointing with a touch-sensitive mouse. *In:* Gross, T., Gulliksen, J., Kotzé, P., Oestreicher, L., Palanque, P., Prates, R. O. and Winckler, M. (Eds.), *Human-Computer Interaction - INTERACT 2009: 12th IFIP TC 13 International Conference*. Uppsala, Sweden, 415–428.

Duchowski, A., 2002. A breadth-first survey of eye-tracking applications. *Behavior Research Methods, Instruments, & Computers*, 34 (4), 455–470.

Duchowski, A. T., Cournia, N. and Murphy, H., 2004. Gaze-contingent displays: A review. *CyberPsychology & Behavior*, 7 (6), 621–634.

Ehmke, C. and Wilson, S., 2007. Identifying web usability problems from eye-tracking data. *In: Proceedings of the 21st British HCI Group Annual Conference on People and Computers: HCI...But Not As We Know It - Volume 1*. Swinton, UK: British Computer Society. BCS-HCI '07, 119–128.

Elmqvist, N. and Tsigas, P., 2007. View-projection animation for 3d occlusion management. *Computers & Graphics*, 31 (6), 864–876.

Elmqvist, N. and Tsigas, P., 2008. A taxonomy of 3d occlusion management for visualization. *IEEE Transactions on Visualization and Computer Graphics*, 14 (5), 1095–1109.

Elmqvist, N. and Tudoreanu, M. E., 2007. Occlusion management in immersive and desktop 3d virtual environments: Theory and evaluation. *IJVR*, 6 (2), 21–32.

Engbert, R. and Kliegl, R., 2003. Microsaccades uncover the orientation of covert attention. *Vision research*, 43 (9), 1035–1045.

Feiner, A. and Steven, O., 2003. The flexible pointer: An interaction technique for selection in augmented and virtual reality. *In: Proc. UIST'03*. 81–82.

Fitts, P. M., 1954. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of experimental psychology*, 47 (6), 381–391.

Forlines, C. and Balakrishnan, R., 2008. Evaluating tactile feedback and direct vs. indirect stylus input in pointing and crossing selection tasks. *In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM. 1563–1572.

Forsberg, A., Herndon, K. and Zeleznik, R., 1996. Aperture based selection for immersive virtual environments. *In: Proceedings of the 9th Annual ACM Symposium on User Interface Software and Technology*. Seattle, Washington, USA, UIST '96, 95–96.

Franke, I. S., Gnther, T. and Groh, R., 2014. *Saccade Detection and Processing for Enhancing 3D Visualizations in Real-Time*. Springer, 317–322.

Frees, S., Kessler, G. D. and Kay, E., 2007. Prism interaction for enhancing control in immersive virtual environments. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 14 (1), 2.

Frisoli, A., Loconsole, C., Leonardis, D., Banno, F., Barsotti, M., Chisari, C. and Bergamasco, M., 2012. A new gaze-bci-driven control of an upper limb exoskeleton for rehabilitation in real-world tasks. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 42 (6), 1169–1179.

Gauthier, G. M., Vercher, J. L., Mussa Ivaldi, F. and Marchetti, E., 1988.

Oculo-manual tracking of visual targets: control learning, coordination control and coordination model. *Experimental Brain Research*, 73 (1), 127–137.

Gielen, C. C. A. M., Dijkstra, T. M. H., Roozen, I. J. and Welten, J., 2009. Coordination of gaze and hand movements for tracking and tracing in 3d. *cortex*, 45 (3), 340–355.

Gowen, E. and Miall, R. C., 2006. Eyehand interactions in tracing and drawing tasks. *Human Movement Science*, 25 (45), 568–585.

Granholm, E. and Steinhauer, S. R., 2004. Pupillometric measures of cognitive and emotional processes. *International Journal of Psychophysiology*, 52 (1), 1–6.

Grossman, T. and Balakrishnan, R., 2006. 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*. Montreux, Switzerland: ACM. UIST '06, 3–12.

Guo, J. and Guo, A., 2005. Crossmodal interactions between olfactory and visual learning in drosophila. *Science*, 309 (5732), 307–310.

Guo, Q. and Agichtein, E., 2010. Towards predicting web searcher gaze position from mouse movements. *In: CHI'10 Extended Abstracts on Human Factors in Computing Systems*. Atlanta, Georgia: ACM. 3601–3606.

Hales, J., Rozado, D. and Mardanbegi, D., . Interacting with objects in the environment by gaze and hand gestures. *In: Proceedings of the 3rd International Workshop on Pervasive Eye Tracking and Mobile Eye-Based Interaction*. 1–9.

Hillaire, S., Lcuyer, A., Cozot, R. and Casiez, G., 2008. Using an eye-tracking system to improve camera motions and depth-of-field blur effects in virtual environments. *In: Virtual Reality Conference, 2008. VR'08*. IEEE. 47–50.

Hincapié-Ramos, J. D., Ozacar, K., Irani, P. P. and Kitamura, Y., 2015. Gyrowand: Imu-based raycasting for augmented reality head-mounted displays. *In: Proceedings of the 3rd ACM Symposium on Spatial User Interaction*. Los Angeles, California, USA: ACM. SUI '15, 89–98.

Irwin, D. E., 1993. *Perceiving an integrated visual world*. Cambridge, MA: MIT Press, 121–142.

Jacob, R. J., 1990. What you look at is what you get: eye movement-based interaction techniques. *In: Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM. 11–18.

Jacobs, L. F. and Schenk, F., 2003. Unpacking the cognitive map: the parallel map theory of hippocampal function. *Psychological review*, 110 (2), 285.

Jalaliniya, S., Mardanbegi, D. and Pederson, T., 2015. Magic pointing for eyewear computers. *In: Proceedings of the 2015 ACM International Symposium on Wearable Computers*. Osaka, Japan: ACM. 155–158.

James, D. R., Leff, D. R., Orihuela-Espina, F., Kwok, K.-W., Mylonas, G. P., Athanasiou, T., Darzi, A. W. and Yang, G.-Z., 2013. Enhanced frontoparietal network architectures following "gaze-contingent" versus "free-hand" motor learning. *NeuroImage*, 64, 267–276.

Jiang, X., Zheng, B., Bednarik, R. and Atkins, M. S., 2015. Pupil responses to continuous aiming movements. *International Journal of Human-Computer Studies*, 83, 1–11.

Johansson, R. S., Westling, G., Bckstrm, A. and Flanagan, J. R., 2001. Eye-hand coordination in object manipulation. *The Journal of neuroscience*, 21 (17), 6917–6932.

Kangas, J., Akkil, D., Rantala, J., Isokoski, P., Majaranta, P. and Raisamo, R., 2014a. Gaze gestures and haptic feedback in mobile devices. *In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Toronto, Ontario, Canada: ACM. 435–438.

Kangas, J., Rantala, J., Akkil, D., Isokoski, P., Majaranta, P. and Raisamo, R., 2014b. Delayed haptic feedback to gaze gestures. *In: Proceedings of*

Haptics: Neuroscience, Devices, Modeling, and Applications: 9th International Conference, EuroHaptics 2014. Versailles, France, 25–31.

Kangas, J., Rantala, J., Akkil, D., Isokoski, P., Majaranta, P. and Raisamo, R., 2017. Vibrotactile stimulation of the head enables faster gaze gestures. *International Journal of Human-Computer Studies*, 98, 62 – 71.

Kapoula, Z. and Robinson, D., 1986. Saccadic undershoot is not inevitable: saccades can be accurate. *Vision research*, 26 (5), 735–743.

Kawashima, T., Terashima, T., Nagasaki, T. and Toda, M., 2005. *Enhancing visual perception using dynamic updating of display*. Springer, 127–141.

Kincaid, J. P. and Westerlund, K. K., 2009. Simulation in education and training. *In: Simulation Conference (WSC), Proceedings of the 2009 Winter*. IEEE. 273–280.

Kirkby, J. A., Webster, L. A., Blythe, H. I. and Liversedge, S. P., 2008. Binocular coordination during reading and non-reading tasks. *Psychological bulletin*, 134 (5), 742.

Kitamura, Y., Yee, A. and Kishino, F., 1998. A sophisticated manipulation aid in a virtual environment using dynamic constraints among object faces. *Presence: Teleoperators and Virtual Environments*, 7 (5), 460–477.

Kleinsmith, A. and Bianchi-Berthouze, N., 2013. Affective body expression perception and recognition: A survey. *IEEE Transactions on Affective Computing*, 4 (1), 15–33.

Kopper, R., Bacim, F. and Bowman, D. A., 2011. Rapid and accurate 3d selection by progressive refinement. *In: 3D User Interfaces (3DUI), 2011 IEEE Symposium on*. IEEE. 67–74.

Kumar, M., Paepcke, A. and Winograd, T., 2007. Eyepoint: Practical pointing and selection using gaze and keyboard. *In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. San Jose, California, USA, CHI '07, 421–430.

Kunkel, N., Soechtig, S., Miniman, J. and Stauch, C., 2016. Augmented and virtual reality go to work: Seeing business through a different lens. *In: Tech Trends 2016: Innovating in the digital era*, Deloitte University Press.

Lankford, C., 2000. Effective eye-gaze input into windows. *In: Proceedings of the 2000 Symposium on Eye Tracking Research & Applications*. Palm Beach Gardens, Florida, USA, ETRA '00, 23–27.

Lavie, N., Hirst, A., de Fockert, J. W. and Viding, E., 2004. Load theory of selective attention and cognitive control. *Journal of Experimental Psychology: General*, 133 (3), 339–354.

Li, W., Ritter, L., Agrawala, M., Curless, B. and Salesin, D., 2007. Interactive cutaway illustrations of complex 3d models. *In: ACM SIGGRAPH 2007 Papers*. San Diego, California, SIGGRAPH '07.

Liang, J. and Green, M., 1994. Jdcad: A highly interactive 3d modeling system. *Computers & Graphics*, 18 (4), 499 – 506.

Liebling, D. J. and Dumais, S. T., 2014. Gaze and mouse coordination in everyday work. *In: Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*. Seattle, Washington: ACM. 1141–1150.

Liversedge, S. P. and Findlay, J. M., 2000. Saccadic eye movements and cognition. *Trends in Cognitive Sciences*, 4 (1), 6–14.

Long, B., Seah, S. A., Carter, T. and Subramanian, S., 2014. Rendering volumetric haptic shapes in mid-air using ultrasound. *ACM Trans. Graph.*, 33 (6), 181:1–181:10.

Mandel, T., 1997. *The Elements of User Interface Design*. New York, NY, USA: John Wiley & Sons, Inc.

Marin, G., Dominio, F. and Zanuttigh, P., 2015. Hand gesture recognition with jointly calibrated leap motion and depth sensor. *Multimedia Tools and Applications*, 1–25.

Masia, L., Casadio, M., Sandini, G. and Morasso, P., 2009. Eye-hand coordination during dynamic visuomotor rotations. *PLoS One*, 4 (9), e7004.

McKee, S. P. and Nakayama, K., 1984. The detection of motion in the peripheral visual field. *Vision research*, 24 (1), 25–32.

McLaughlin, A. C., Rogers, W. A. and Fisk, A. D., 2009. Using direct and indirect input devices: Attention demands and age-related differences. *ACM transactions on computer-human interaction : a publication of the Association for Computing Machinery*, 16 (1), 1–15.

Mine, M., 1995. *Virtual environment interaction techniques*. Technical Report TR95-018, UNC Chapel Hill computer science.

Miniotas, D., Špakov, O. and MacKenzie, I. S., 2004. Eye gaze interaction with expanding targets. *In: CHI '04 Extended Abstracts on Human Factors in Computing Systems*. Vienna, Austria, CHI EA '04, 1255–1258.

Monden, A., Matsumoto, K.-i. and Yamato, M., 2005. Evaluation of gaze-added target selection methods suitable for general guis. *International journal of computer applications in technology*, 24 (1), 17–24.

Moreno, R. and Mayer, R., 2007. Interactive multimodal learning environments. *Educational Psychology Review*, 19 (3), 309–326.

Mrotek, L., Gielen, C. C. A. M. and Flanders, M., 2006. Manual tracking in three dimensions. *Experimental Brain Research*, 171 (1), 99–115.

Murphy, H. A., Duchowski, A. T. and Tyrrell, R. A., 2009. Hybrid image/model-based gaze-contingent rendering. *ACM Trans. Appl. Percept.*, 5 (4), 1–21.

Mylonas, G. P., Kwok, K.-W., James, D. R. C., Leff, D., Orihuela-Espina, F., Darzi, A. and Yang, G.-Z., 2012. Gaze-contingent motor channelling, haptic constraints and associated cognitive demand for robotic mis. *Medical Image Analysis*, 16 (3), 612–631.

Neggers, S. F. and Bekkering, H., 2000. Ocular gaze is anchored to the target of an ongoing pointing movement. *Journal of Neurophysiology*, 83 (2), 639–651.

Nielsen, J., 1994. *Usability engineering*. Elsevier.

Oh, J.-Y. and Stuerzlinger, W., 2004. A system for desktop conceptual 3d design. *Virtual Reality*, 7 (3-4), 198–211.

Okoe, M., Alam, S. S. and Jianu, R., 2014. A gaze-enabled graph visualization to improve graph reading tasks. *Comput. Graph. Forum*, 33 (3), 251–260.

Oviatt, S., 2012. Multimodal interfaces. *In:* Jacko, J. A. (Ed.), *Human computer interaction handbook: Fundamentals, evolving technologies, and emerging applications*, CRC press, chapter 18, 405.

Oviatt, S., Cohen, P., Wu, L., Duncan, L., Suhm, B., Bers, J., Holzman, T., Winograd, T., Landay, J., Larson, J. and Ferro, D., 2000. Designing the user interface for multimodal speech and pen-based gesture applications: State-of-the-art systems and future research directions. *HumanComputer Interaction*, 15 (4), 263–322.

Palinko, O., Kun, A. L., Shyrokov, A. and Heeman, P., 2010. Estimating cognitive load using remote eye tracking in a driving simulator. *In: Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications*. ACM. 141–144.

Pashler, H., 2013. *Encyclopedia of the Mind*, volume 1. SAGE.

Paulson, D. S., 2007. *Handbook of regression and modeling: Applications for the clinical and pharmaceutical industries*. Boca Raton, FL: Chapman & Hall/CRC.

Periverzov, F. and Ilies, H., 2015. Ids: The intent driven selection method for natural user interfaces. *In: 3D User Interfaces (3DUI), 2015 IEEE Symposium on*. 121–128.

Pfeiffer, T., Latoschik, M. E. and Wachsmuth, I., 2008. Evaluation of binocular eye trackers and algorithms for 3d gaze interaction in virtual reality environments. *JVRB-Journal of Virtual Reality and Broadcasting*, 5 (16).

Pfeuffer, K., Alexander, J., Chong, M. K. and Gellersen, H., 2014. Gaze-touch: combining gaze with multi-touch for interaction on the same surface. *In: Proceedings of the 27th annual ACM symposium on User interface software and technology*. Honolulu, Hawaii, USA: ACM. 509–518.

Pfeuffer, K., Alexander, J., Chong, M. K., Zhang, Y. and Gellersen, H., 2015. Gaze-shifting: Direct-indirect input with pen and touch modulated by gaze. *In: Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*. Daegu, Kyungpook, Republic of Korea: ACM. 373–383.

Pfeuffer, K., Vidal, M., Turner, J., Bulling, A. and Gellersen, H., 2013. Pursuit calibration: Making gaze calibration less tedious and more flexible. *In: Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology*. ACM. UIST '13, 261–270.

Pierce, J. S., Forsberg, A. S., Conway, M. J., Hong, S., Zeleznik, R. C. and Mine, M. R., 1997. Image plane interaction techniques in 3d immersive environments. *In: Proceedings of the 1997 Symposium on Interactive 3D Graphics*. Providence, Rhode Island, USA: ACM. I3D '97, 39–ff.

Poole, A. and Ball, L. J., 2006. Eye tracking in hci and usability research. *Encyclopedia of human computer interaction*, 1, 211–219.

Posner, M. I., 1980. Orienting of attention. *Quarterly journal of experimental psychology*, 32 (1), 3–25.

Pouke, M., Karhu, A., Hickey, S. and Arhippainen, L., 2012. Gaze tracking and non-touch gesture based interaction method for mobile 3d virtual spaces. *In: Proceedings of the 24th Australian Computer-Human Interaction Conference*. ACM. 505–512.

Poupyrev, I., Billinghurst, M., Weghorst, S. and Ichikawa, T., 1996. The go-go interaction technique: non-linear mapping for direct manipulation in vr. *In: Proceedings of the 9th annual ACM symposium on User interface software and technology*. Seattle, Washington: ACM. 79–80.

Poupyrev, I. and Ichikawa, T., 1999. Manipulating objects in virtual worlds: Categorization and empirical evaluation of interaction techniques. *Journal of Visual Languages & Computing*, 10 (1), 19 – 35.

Purves, D., Augustine, G. J., Fitzpatrick, D., Katz, L. C., LaMantia, A.-S., McNamara, J. O. and Williams, S. M. (Eds.), 2001. Types of eye movements and their functions. *Neuroscience*. 2nd edition. Sunderland, MA: Sinauer Associates.

Rantala, J., Kangas, J., Akkil, D., Isokoski, P. and Raisamo, R., 2014. Glasses with haptic feedback of gaze gestures. *In: CHI '14 Extended Abstracts on Human Factors in Computing Systems*. Toronto, Ontario, Canada: ACM. 1597–1602.

Rayner, K., 1998. Eye movements in reading and information processing: 20 years of research. *Psychological bulletin*, 124 (3), 372–422.

Rayner, K., 2009. Eye movements and attention in reading, scene perception, and visual search. *The quarterly journal of experimental psychology*, 62 (8), 1457–1506.

Reddy, M., 1998. Specification and evaluation of level of detail selection criteria. *Virtual Reality*, 3 (2), 132–143.

Reichelt, S., Häussler, R., Fütterer, G. and Leister, N., 2010. Depth cues in human visual perception and their realization in 3d displays. *In: SPIE Defense, Security, and Sensing*. International Society for Optics and Photonics. 76900B–76900B.

Reina, G. A. and Schwartz, A. B., 2003. Eyehand coupling during closed-loop drawing: Evidence of shared motor planning? *Human Movement Science*, 22 (2), 137–152.

Reingold, E. M., Loschky, L. C., McConkie, G. W. and Stampe, D. M., 2003. Gaze-contingent multiresolutional displays: An integrative review. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 45 (2), 307–328.

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

Rodden, K., Fu, X., Aula, A. and Spiro, I., 2008. Eye-mouse coordination patterns on web search results pages. *In: CHI '08 Extended Abstracts on Human Factors in Computing Systems*. Florence, Italy: ACM. 2997–3002.

Rosenberg, L. B., 1993. Virtual fixtures: Perceptual tools for telerobotic manipulation. *In: Virtual Reality Annual International Symposium, 1993., 1993 IEEE*. IEEE. 76–82.

Rottach, K. G., Zivotofsky, A. Z., Das, V. E., Averbuch-Heller, L. E. A., Discenna, A. O., Poonyathalang, A. and Leigh, R. J., 1996. Comparison of horizontal, vertical and diagonal smooth pursuit eye movements in normal human subjects. *Vision research*, 36 (14), 2189–2195.

Rozado, D., 2013. Mouse and keyboard cursor warping to accelerate and reduce the effort of routine hci input tasks. *IEEE Transactions on Human-Machine Systems*, 43 (5), 487–493.

Ruhland, K., Peters, C. E., Andrist, S., Badler, J. B., Badler, N. I., Gleicher, M., Mutlu, B. and McDonnell, R., 2015. A review of eye gaze in virtual agents, social robotics and hci: Behaviour generation, user interaction and perception. *Computer Graphics Forum*, 34 (6), 299–326.

Salvucci, D. D. and Anderson, J. R., 2000. Intelligent gaze-added interfaces. *In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. The Hague, The Netherlands, CHI '00, 273–280.

Sand, A., Rakkolainen, I., Isokoski, P., Kangas, J., Raisamo, R. and Palovuori, K., 2015. Head-mounted display with mid-air tactile feedback. *In: Proceedings of the 21st ACM Symposium on Virtual Reality Software and Technology*. ACM. VRST '15, 51–58.

Sauro, J., 2013. *10 Things To Know About The System Usability Scale (SUS)* [online]. Available from: http://www.measuringu.com/blog/10-things-SUS.php [Accessed 9 March 2016].

Savitzky, A. and Golay, M. J., 1964. Smoothing and differentiation of data by simplified least squares procedures. *Analytical chemistry*, 36 (8), 1627–1639.

Scheggi, S., Meli, L., Pacchierotti, C. and Prattichizzo, D., 2015. Touch the virtual reality: Using the leap motion controller for hand tracking and wearable tactile devices for immersive haptic rendering. *In: ACM SIGGRAPH 2015 Posters*. Los Angeles, California: ACM. SIGGRAPH '15, 31:1–31:1.

Schultheis, H. and Jameson, A., 2004. Assessing cognitive load in adaptive hypermedia systems: Physiological and behavioral methods. *In: Adaptive hypermedia and adaptive web-based systems*. Springer. 225–234.

Schumacher, J., Allison, R. and Herpers, R., 2004. Using saccadic suppression to hide graphic updates. *In: Eurographic/ACM SIGGRAPH Symposium on Virtual Environments*. 17–24.

Sebe, N., 2009. Multimodal interfaces: Challenges and perspectives. *Journal of Ambient Intelligence and smart environments*, 1 (1), 23–30.

Shneiderman, B., 1991. Touch screens now offer compelling uses. *IEEE software*, 8 (2), 93–94.

Simeone, A. L., 2016. Indirect touch manipulation for interaction with stereoscopic displays. *In: 3D User Interfaces (3DUI), 2016 IEEE Symposium on*. IEEE. 13–22.

Simeone, A. L., Bulling, A., Alexander, J. and Gellersen, H., 2016. Three-point interaction: Combining bi-manual direct touch with gaze. *In: Proceedings of the International Working Conference on Advanced Visual Interfaces*. Bari, Italy: ACM. 168–175.

Simeone, A. L. and Gellerseny, H., 2015. Comparing indirect and direct touch in a stereoscopic interaction task. *In: 3D User Interfaces (3DUI), 2015 IEEE Symposium on*. IEEE. 105–108.

Singh, K. and Balakrishnan, R., 2004. Visualizing 3d scenes using non-linear projections and data mining of previous camera movements. *In: International Conference on Computer Graphics, Virtual Reality, Visualisation and Interaction in Africa, Afrigraph 2004*. Stellenbosch, South Africa, 41–48.

Slambekova, D., Bailey, R. and Geigel, J., 2012. Gaze and gesture based object manipulation in virtual worlds. *In: Proceedings of the 18th ACM symposium on Virtual reality software and technology*. Toronto, Ontario, Canada: ACM. 203–204.

Smith, B. A., Ho, J., Ark, W. and Zhai, S., 2000. Hand eye coordination patterns in target selection. *In: Proceedings of the 2000 symposium on Eye tracking research & applications*. Palm Beach Gardens, Florida: ACM. 117–122.

Sodhi, R., Poupyrev, I., Glisson, M. and Israr, A., 2013. Aireal: interactive tactile experiences in free air. *ACM Transactions on Graphics (TOG)*, 32 (4), 134.

Song, J., Cho, S., Baek, S.-Y., Lee, K. and Bang, H., 2014. Gafinc: Gaze and finger control interface for 3d model manipulation in cad application. *Computer-Aided Design*, 46 (0), 239–245.

Steed, A., 2006. Towards a general model for selection in virtual environments. *In: 3D User Interfaces (3DUI'06)*. 103–110.

Steed, A. and Parker, C., 2004. 3d selection strategies for head tracked and non-head tracked operation of spatially immersive displays. *In: 8th International Immersive Projection Technology Workshop*. 13–14.

Stein, B. E. and Meredith, M. A., 1993. *The merging of the senses.* The MIT Press.

Stellmach, S. and Dachselt, R., 2012. Look & touch: gaze-supported target acquisition. *In: Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems*. Austin, Texas: ACM. 2981–2990.

Stellmach, S. and Dachselt, R., 2013. Still looking: Investigating seamless gaze-supported selection, positioning, and manipulation of distant targets. *In: Proceedings of the 2013 SIGCHI Conference on Human Factors in Computing Systems*. ACM. 285–294.

Stoakley, R., Conway, M. J. and Pausch, R., 1995. Virtual reality on a wim:

interactive worlds in miniature. *In: Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM Press/Addison-Wesley Publishing Co. 265–272.

Sundstedt, V., Stavrakis, E., Wimmer, M. and Reinhard, E., 2008. A psychophysical study of fixation behavior in a computer game. *In: Proceedings of the 5th symposium on Applied perception in graphics and visualization*. Los Angeles, California: ACM. 43–50.

Tanriverdi, V. and Jacob, R. J. K., 2000. Interacting with eye movements in virtual environments. *In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. The Hague, The Netherlands: ACM. CHI '00, 265–272.

Tchalenko, J. and Miall, C. R., 2009. Eyehand strategies in copying complex lines. *cortex*, 45 (3), 368–376.

Templin, K., Didyk, P., Myszkowski, K., Hefeeda, M. M., Seidel, H.-P. and Matusik, W., 2014. Modeling and optimizing eye vergence response to stereoscopic cuts. *ACM Trans. Graph.*, 33 (4), 145:1–145:8.

Minority Report, 2002. [film still]. Directed by Steven Spielberg. USA: Twentieth Century Fox Film Corporation and Dreamworks. Available from: http://www.businessinsider.com.au/minority-report-interface-offices-2015-11 [Accessed 6 June 2016].

Tramper, J. J. and Gielen, C., 2011. Visuomotor coordination is different for different directions in three-dimensional space. *The Journal of neuroscience*, 31 (21), 7857–7866.

Triesch, J., Sullivan, B. T., Hayhoe, M. M. and Ballard, D. H., 2002. Saccade contingent updating in virtual reality. *In: Proceedings of the 2002 symposium on Eye tracking research & applications*. ACM. 95–102.

Tse, P. U., Sheinberg, D. S. and Logothetis, N. K., 2004. The distribution of microsaccade directions need not reveal the location of attention: reply to rolfs, engbert, and kliegl. *Psychological Science*, 15 (10), 708–710.

Turk, M., 2014. Multimodal interaction: A review. *Pattern Recognition Letters*, 36, 189–195.

Turner, J., Alexander, J., Bulling, A., Schmidt, D. and Gellersen, H., 2013. Eye pull, eye push: Moving objects between large screens and personal devices with gaze and touch. *In:* Kotzé, P., Marsden, G., Lindgaard, G., Wesson, J. and Winckler, M. (Eds.), *Human-Computer Interaction - INTERACT 2013: 14th IFIP TC 13 International Conference*. Cape Town, South Africa: Springer Berlin Heidelberg. 170–186.

Velloso, E., Turner, J., Alexander, J., Bulling, A. and Gellersen, H., 2015. An empirical investigation of gaze selection in mid-air gestural 3d manipulation. *In:* Abascal, J., Barbosa, S., Fetter, M., Gross, T., Palanque, P. and Winckler, M. (Eds.), *Human-Computer Interaction – INTERACT 2015: 15th IFIP TC 13 International Conference, Bamberg, Germany, September 14-18, 2015, Proceedings, Part II*. Cham: Springer International Publishing. 315–330.

Velloso, E., Wirth, M., Weichel, C., Esteves, A. and Gellersen, H., 2016. Ambigaze: Direct control of ambient devices by gaze. *In: Proceedings of the 2016 ACM Conference on Designing Interactive Systems*. ACM. DIS '16, 812–817.

Vercher, J. L. and Gauthier, G. M., 1992. Oculo-manual coordination control: Ocular and manual tracking of visual targets with delayed visual feedback of the hand motion. *Experimental Brain Research*, 90 (3), 599–609.

Vidal, M., Bulling, A. and Gellersen, H., 2013. Pursuits: spontaneous interaction with displays based on smooth pursuit eye movement and moving targets. *In: Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computing*. Zurich, Switzerland: ACM. 439–448.

Vlaskamp, B. N. and Hooge, I. T. C., 2006. Crowding degrades saccadic search performance. *Vision research*, 46 (3), 417–425.

Vogel, D. and Balakrishnan, R., 2005. 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. Seattle, WA, USA: ACM. UIST '05, 33–42.

Špakov, O., 2011. Comparison of gaze-to-objects mapping algorithms. *In: Proceedings of the 1st Conference on Novel Gaze-Controlled Applications*. Karlskrona, Sweden, 1–8.

Wang, Y. and MacKenzie, C. L., 1999. Effects of orientation disparity between haptic and graphic displays of objects in virtual environments. *In:* Sasse, M. A. and Johnson, C. (Eds.), *Proc. of the 15th IFIP TC13 Conference on Human-Computer Interaction (INTERACT '99)*. Edinburgh, UK: IOS Press. 391–398.

Wang Baldonado, M. Q., Woodruff, A. and Kuchinsky, A., 2000. Guidelines for using multiple views in information visualization. *In: Proceedings of the Working Conference on Advanced Visual Interfaces*. Palermo, Italy, AVI '00, 110–119.

Watanabe, J., Ando, H., Maeda, T. and Tachi, S., 2007. Gaze-contingent visual presentation based on remote saccade detection. *Presence: Teleoperators and Virtual Environments*, 16 (2), 224–234.

Watanabe, J., Maeda, T. and Ando, H., 2012. Gaze-contingent visual presentation technique with electro-ocular-graph-based saccade detection. *ACM Transactions on Applied Perception (TAP)*, 9 (2), 6.

Weibelzahl, S. and Weber, G., 2002. Advantages, opportunities and limits of empirical evaluations: Evaluating adaptive systems. *KI*, 16 (3), 17–20.

Wonner, J., Grosjean, J., Capobianco, A. and Bechmann, D., 2012. Starfish: a selection technique for dense virtual environments. *In: Proceedings of the 18th ACM symposium on Virtual reality software and technology*. ACM. 101–104.

Wu, H. and Wang, J., 2016. A visual attention-based method to address the midas touch problem existing in gesture-based interaction. *The Visual Computer*, 32 (1), 123–136.

Wyss, H. P., Blach, R. and Bues, M., 2006. isith - intersection-based spatial interaction for two hands. *In: 3D User Interfaces (3DUI'06)*. 59–61.

Xia, R. and Barnes, G., 1999. Oculomanual coordination in tracking of pseudorandom target motion stimuli. *Journal of Motor Behavior*, 31 (1), 21–38.

Yamaguchi, S., Konishi, K., Yasunaga, T., Yoshida, D., Kinjo, N., Kobayashi, K., Ieiri, S., Okazaki, K., Nakashima, H. and Tanoue, K., 2007. Construct validity for eyehand coordination skill on a virtual reality laparoscopic surgical simulator. *Surgical endoscopy*, 21 (12), 2253–2257.

Yoo, B., Han, J.-J., Choi, C., Yi, K., Suh, S., Park, D. and Kim, C., 2010. 3d user interface combining gaze and hand gestures for large-scale display. *In: CHI'10 Extended Abstracts on Human Factors in Computing Systems*. ACM. 3709–3714.

Zhai, S., Morimoto, C. and Ihde, S., 1999. Manual and gaze input cascaded (MAGIC) pointing. *In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Pittsburgh, Pennsylvania: ACM. 246–253.

Zhang, Y., Stellmach, S., Sellen, A. and Blake, A., 2015. The costs and benefits of combining gaze and hand gestures for remote interaction. *In:* Abascal, J., Barbosa, S., Fetter, M., Gross, T., Palanque, P. and Winckler, M. (Eds.), *Human-Computer Interaction INTERACT 2015: 15th IFIP TC 13 International Conference, Bamberg, Germany, September 14-18, 2015, Proceedings, Part III*. Springer International Publishing. 570–577.

Zhu, D., Gedeon, T. and Taylor, K., 2011. "moving to the centre": A gaze-driven remote camera control for teleoperation. *Interacting with Computers*, 23 (1), 85–95.

Appendix A

More details of the Scaling scheme

The scaling is not linear because the scaling factor depends on the moving direction. The position of the target object is important only because it defines the moving direction. Figure A.1 illustrates and explains how the scaling scheme works using the example in Figure 4.4. As we explained earlier in Chapter 4, once a cube is selected, a displacement will be generated which causes the mismatch of the virtual space and the physical space (see Figure A.1a). At this moment, the virtual hand can only operate inside the intersection of the two spaces, marked as the shaded rectangle in Figure A.1a. To make sure the virtual hand can still reach all positions inside the original virtual space, we need to remap the intersection area to the whole virtual space. The grey area S does not need scaling because this area is equivalent to the corresponding area in the virtual space so the virtual hand can still reach every position inside it. The rest of the intersection needs to be mapped to the rest of the original virtual space as shown in Figure A.1b. It is clear that on different direction the scaling factor is different.

Figure A.2 illustrates two extreme cases. In Figure A.2a, the virtual hand can freely reach everywhere inside the grey area S without scaling, but it cannot reach the rest of the virtual space because the starting position of the hand is at the corner, there is no room for it to move backwards or leftwards. In Figure A.2b, the scaling area of the intersection is very tiny which will be mapped to a very large area (the whole virtual space minus the grey area S)

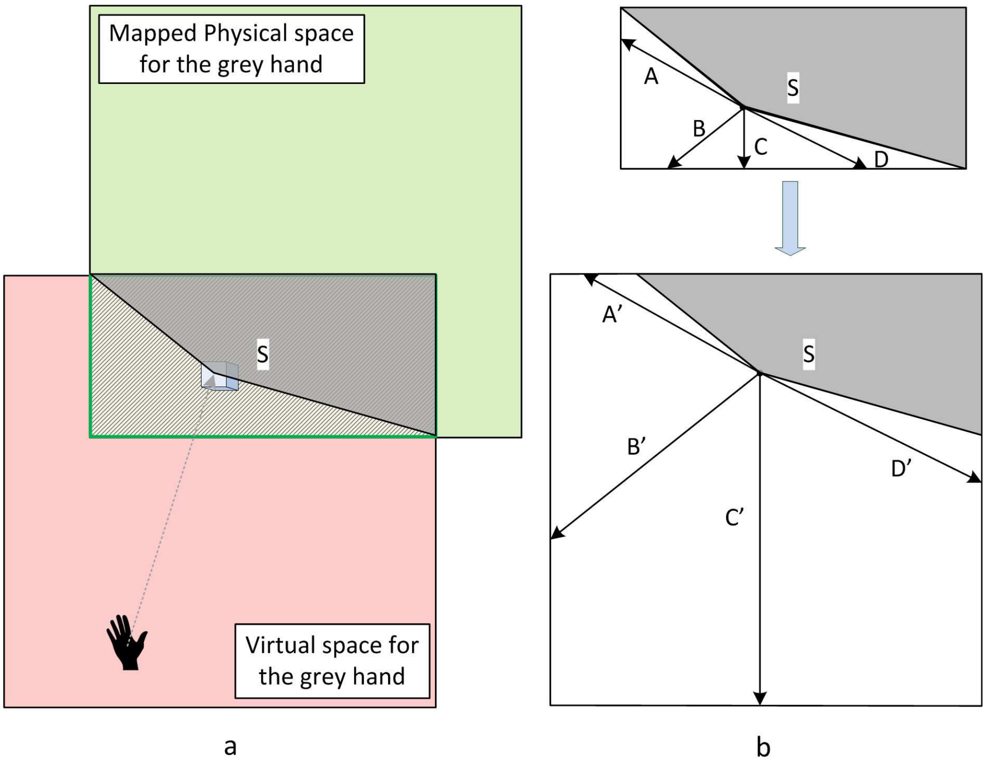

Mapped Physical space for the grey hand
S
Virtual space for the grey hand
a
A
B
C
D
S
A'
B'
C'
D'
S
b

Figure A.1: *Explanation of the Scaling method.*

so the scaling factor will be very large. If the hand move a little more down to the boundary, then there will be no intersection between the two spaces hence the cube can be hardly moved, it will either be blocked by the upper virtual wall, or the hand will go outside of the lower boundary.

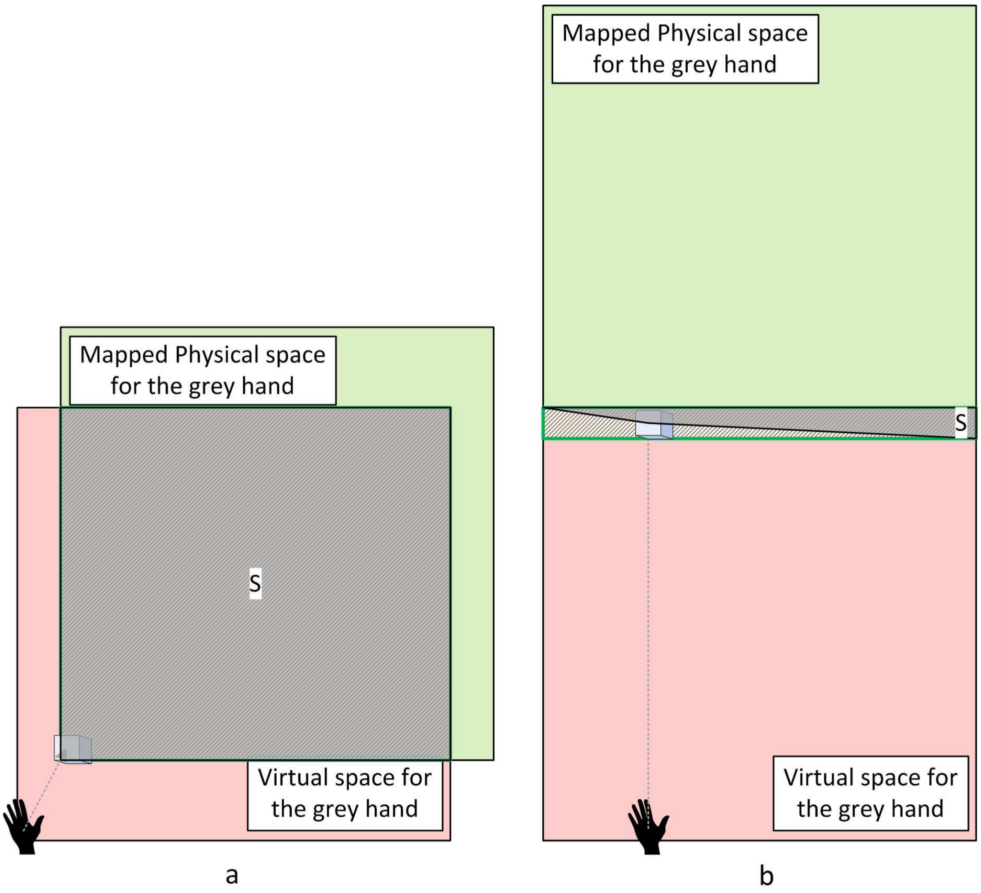
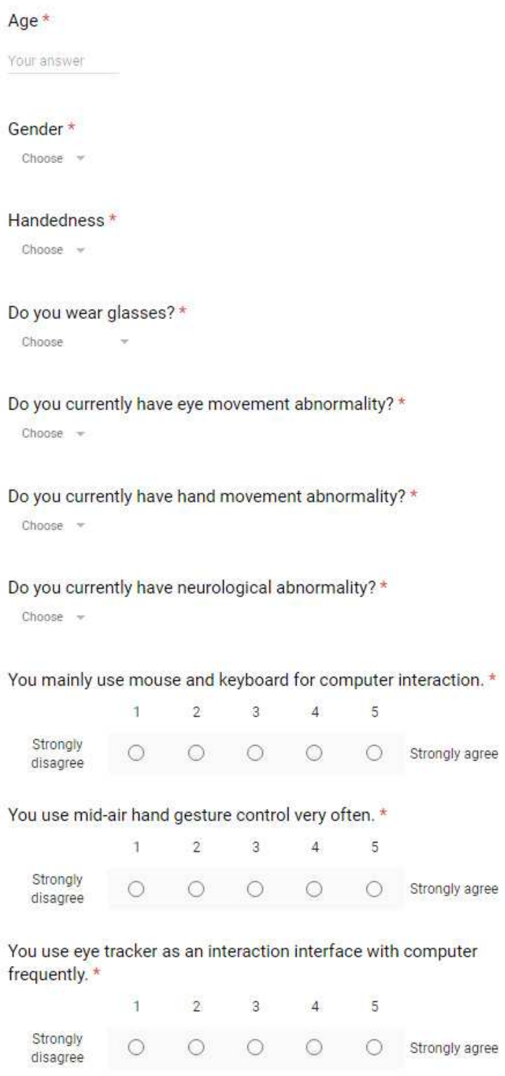

Mapped Physical space for the grey hand
S
Virtual space for the grey hand
a
Mapped Physical space for the grey hand
S
Virtual space for the grey hand
b

Figure A.2: *Two extreme cases of the Scaling method.*

Appendix B

Questionnaire



Age *

Your answer

Gender *

Choose

Handedness *

Choose

Do you wear glasses? *

Choose

Do you currently have eye movement abnormality? *

Choose

Do you currently have hand movement abnormality? *

Choose

Do you currently have neurological abnormality? *

Choose

You mainly use mouse and keyboard for computer interaction. *

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

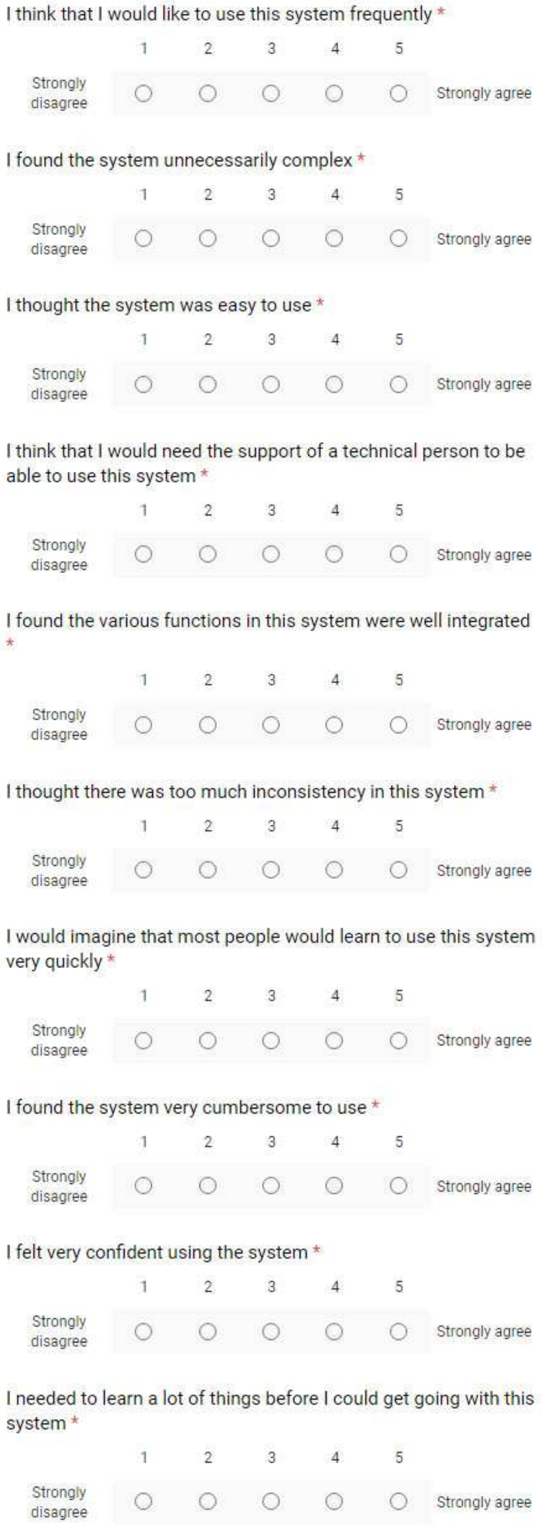
You use mid-air hand gesture control very often. *

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

You use eye tracker as an interaction interface with computer frequently. *

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

Figure B.1: *The demographic questionnaire.*



I think that I would like to use this system frequently *

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

I found the system unnecessarily complex *

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

I thought the system was easy to use *

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

I think that I would need the support of a technical person to be able to use this system *

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

I found the various functions in this system were well integrated *

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

I thought there was too much inconsistency in this system *

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

I would imagine that most people would learn to use this system very quickly *

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

I found the system very cumbersome to use *

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

I felt very confident using the system *

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

I needed to learn a lot of things before I could get going with this system *

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

Figure B.2: *The SUS questionnaire.*