[1]Understanding the impact of multimodal interaction using gaze informed mid-air gesture control in 3D virtual objects manipulation

Shujie Deng[a], Nan Jiang[b], Jian Chang[a], Shihui Guo[c*], Jian J. Zhang[a]

[a] National Centre for Computer Animation, Bournemouth University, Poole, United Kingdom

[b] Department of Computing and Informatics, Bournemouth University, Poole, United Kingdom

[c] School of Software, Xiamen University, Xiamen, China

* Corresponding author

E-mail: guoshihui@xmu.edu.cn

[1] *This is an Accepted Manuscript of an article published by Elsevier in International Journal of Human - Computer Studies on 27th April 2017, available online:* http://dx.doi.org/10.1016/j.ijhcs.2017.04.002

Understanding the impact of multimodal interaction using gaze informed mid-air gesture control in 3D virtual objects manipulation

ABSTRACT

Multimodal interactions provide users with more natural ways to manipulate virtual 3D objects than using traditional input methods. An emerging approach is gaze modulated pointing, which enables users to perform object selection and manipulation in a virtual space conveniently through the use of a combination of gaze and other interaction techniques (e.g., mid-air gestures). As gaze modulated pointing uses different sensors to track and detect user behaviours, its performance relies on the user's 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 user's perception, manipulation errors (e.g., out of boundary errors, proximity errors) may occur. Therefore, in gaze modulated pointing, as gaze can introduce misalignment of the spatial mapping, it may lead to user's misperception of the virtual environment and consequently manipulation errors. This paper provides a clear definition of the problem through a thorough investigation on its causes and specifies the conditions when it occurs, which is further validated in the experiment. It also proposes three methods (Scaling, Magnet and Dual-gaze) to address the problem and examines them using a comparative study which involves 20 participants with 1,040 runs. The results show that all three methods improved the manipulation performance with regard to the defined problem where Magnet and Dual-gaze delivered better performance than Scaling. This finding could be used to inform a more robust multimodal interface design supported by both eye tracking and mid-air gesture control without losing efficiency and stability.

Keywords

Eye tracking; mid-air gesture; 3D interaction; spatial misperception; multimodal interfaces; virtual reality

1. INTRODUCTION

Immersive user experience is one of the fundamental requirements in Virtual Reality (VR) and Augmented Reality (AR) applications (Burdea et al., 1996). This is usually achieved using sensory technologies to realise multimodal interactions which enable users to employ natural modes of communication including voice, gesture, eye tracking, body movement, etc. (Cohen et al., 1997; Quek et al., 2002).

Gaze+gesture is an emerging multimodal technique which allows users to manipulate 3D virtual objects via gaze modulated pointing using a combination of eye tracking and gesture control. In comparison with traditional input methods such as mouse and keyboard, this technique features faster target acquisition with richer control capabilities in all dimensions through the use of eye tracker and depth sensor (Chatterjee et al., 2015; Velloso et al., 2015). The working principle of the Gaze+gesture technique is illustrated in Fig. 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.

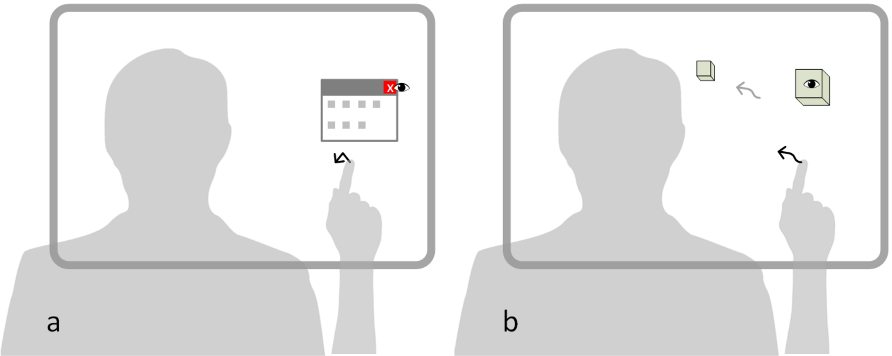

Fig. 1. Two examples of how the Gaze+gesture technique works in typical scenarios. (a) A 2D example of discrete action. To close the window, the user first looks at the "X" button on the upper right corner of the window and then makes a tapping gesture to close it. The user can tap their hand anywhere 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 Fig 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.

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

The results show that the interruption rate of the problem condition has reached 99% which validates our definition. Moreover, although the comparative study indicates that all three approaches can improve usability in the problem condition to different extents, the last two (Magnet and Dual-gaze) provide better performance. Not only addressing a timely problem that is likely to become ever more important as multimodal interaction becomes ubiquitous, the different preference of each method can also help provide guidance for future interaction designs.

The paper is outlined as follows. We discuss the background of our study and how our work is different from existing multimodal interaction studies in Section 2. Following that, the problem is defined in Section 3. We then discuss the strategies to tackle the problem and propose three methods to resolve this issue in Section 4. The experiment is designed to validate the problem definition, which is elaborated in Section 5. The results presented in Section 6 confirm the proposed problem definition and reveal insights of the three proposed methods whose advantages and disadvantages are discussed in Section 7. Followed by design implications in Section 8 we conclude the paper in Section 9 with the future work.

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, we first discuss whether existing interaction techniques have the risk of having this problem based on the three conditions.

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. The relative pointing devices do not strictly require a fixed mapping and Control-Display (CD) gain which, however, are essential to the absolute pointing devices. 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).

A computer mouse is a typical relative pointing device that the relative position of the user's hand/mouse and the mouse cursor is not fixed. 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 with 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. 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 hand to achieve fine selection. This technique has been further improved in many ways such as the cursor setoff constraints (Zhai et al., 1999) and the dynamic eye tracking accuracy (Fares et al., 2013).

Some interaction techniques have the tracking devices or wearable sensors attached to the users’ hand or body, 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. Pouke et al. (2012) combined eye tracker and mid-air gesture interaction using a 6-DOF accelerometer/gyro sensor attached to the user’s hand to perform gesture control. The eye gaze was used for object selection and no cursor of the hand tracker was mentioned particularly. This technique supports selection, translation and rotation, so both discrete and continuous manipulations are available. Similar to a computer mice, this gesture sensor was attached to the user’s hand, so the gesture detection area was always centred on the physical hand position; also it was not a camera-based tracker, so no absolute spatial mapping was involved. This design 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. However, attaching a device to the hand will increase arm fatigue especially for mid-air manipulation because the user needs to constantly hold their hand in the air. 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 more sensitive to the tracking range.

Touch screen and certain depth sensors for gestural control are absolute pointing devices, i.e., the relative position of the hand and its mapped position in the virtual space is fixed. 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 2.2 and 2.3.

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 (McLaughlin et al., 2009).

Direct devices input 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. The explicit presentation of a mobile phone 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. Pfeuffer et al. discussed four 2D Gaze-touch applications, image gallery, paint, map navigation and multiple objects manipulation. Various interaction techniques were applied in the applications. 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 colour but just looking at the colour then tap anywhere on the screen. A three-point interaction technique that combines bi-manual direct touch and gaze was investigated by Simeone et al. (2016). Because of the clearly presented screen boundary, the users could well perceive the working boundaries, so these manipulations were problem free.

Indirect devices do not input body movement into the system equivalently as what the direct input devices do. Instead, a transform from the body coordination to the system coordination is introduced. 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 transform because a great CD gain can result in that a large movement of the device only corresponds to a small movement of the cursor and vice versa. All the components that involve in the transform are integrated into our brain to build a mental representation of the transform.

The touch screens have also been used as indirect input devices such as the external manipulation device in distant displays. Stellmach and Dachselt (2012) designed the Look & Touch technique for 2D object selection on remote displays 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. Simeone (2016) compared the performance of direct and indirect touch in stereoscopic displays. 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.

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.

A discrete action refers to a single hand motion, such as pinching, swiping and grabbing. It is commonly associated with giving a command or sending a confirmation to trigger an action that can be done by the computer itself, for example, pressing a button to close a window (Fig. 1a) or swiping to see the next photo. 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. Thus the tracking range will not be a problem for these implementations. For example, 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 is discrete, such as push and pull for zooming in and out. Hales et al. (2013) 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. This also does not require continuous eye hand coordination. Song et al. (2014) discussed a computer-aided design (CAD) application that used hand gesture for basic manipulation such as translation, zoom and rotation. The application only applied eye tracker to assist zoom by using the gaze position as the centre of zooming. The gesture for zoom command does not require continuous eye hand coordination either.

Continuous manipulation involves constant positional changes in three dimensions, such as dragging (Fig. 1b) and panning. This type of manipulation always initiates and ends with a discrete hand action, respectively. Between the two discrete hand actions, there is a 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 paper is limited to the absolute pointing devices using indirect input methods during continuous manipulation.

2.4. Absolute indirect devices with continuous manipulation

Using eye trackers to modulate selection and translation tasks with absolute indirect gestural devices starts to emerge due to the recent development of cost-effective eye trackers and depth sensors. Except for the studies of Velloso et al. (2015) and Chatterjee et al. (2015) mentioned earlier, Slambekova et al. (2012) reported a framework using a "look at" mechanism for choosing objects, namely that hand gesture was 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. Similar to one of our methods, Dual-gaze, the eye gaze was also used to locate a target position for translation, but they reported with no further details. 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.

Stylus-based haptics is another absolute indirect device. Gaze selection has also been integrated into haptic interactions. Mylonas et al. (2012) proposed two related methods, Gaze-Contingent Motor Channelling (GCMC) and Gaze-Contingent Haptic Constraints (GCHC). GCMC describes the concept that a dynamical force exerts from the haptic tooltip towards the position of gaze in planar manual tracking while GCHC further 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. 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 show improved concentration on task learning quality of novices when force feedback was involved.

After clarifying the premises of the problem, we explain how it occurs in the next section.

3. SPATIAL MISPERCEPTION PROBLEM

In this section, we first review how the conventional Gaze+gesture technique works in an object drag-and-drop example, then we 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.

3.1. Gaze+gesture interaction technique (Normal method)

We implemented a prototype similar to the Gaze+gesture interaction technique presented in previous work (Chatterjee et al., 2015), which we refer to as the Normal method in this paper 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 Fig. 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.

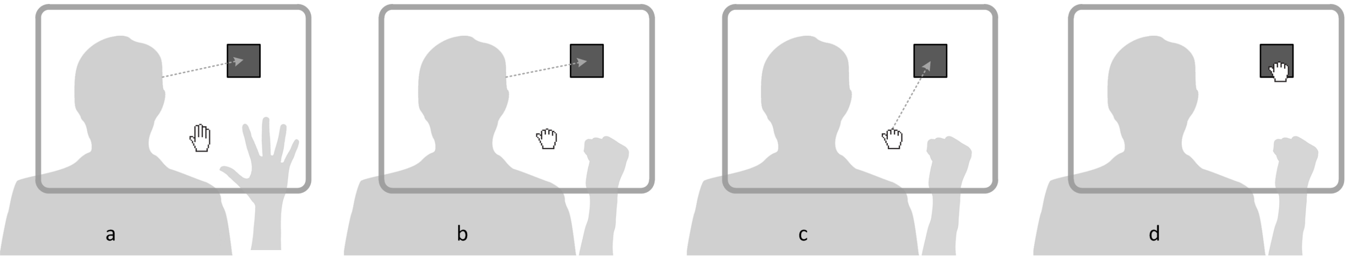

Fig. 2. Illustration of the selection phase of the Normal method. (a) The user is looking at the object to select with hand standby. (b) He makes the grabbing gesture to confirm the selection. (c) After the confirmation, the user does not need to stare at the object anymore. The graphic virtual hand shifts to where the object is, while the physical hand does not move. (d) The graphic virtual hand is shown grabbing the object.

3.2. Interactive interruption

Interruptions were observed in interactions using the Normal method as illustrated in Fig. 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?

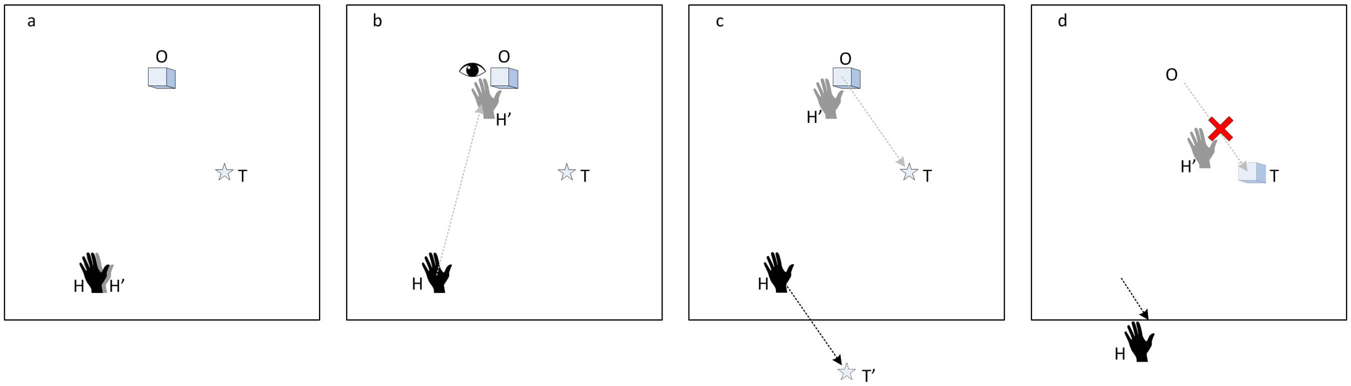

Fig. 3. A case of interrupted translation caused by mapping misperception. (a) The initial scene. The virtual space and the

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.

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 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 minimize or prevent such problems.. Therefore, we give a definition of the spatial misperception problem in the next section.

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.

Fig. 4 gives an illustration of the problem definition using the same example in Fig. 3. Note that the problem we 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 we explain a cause introduced by multimodal integration.

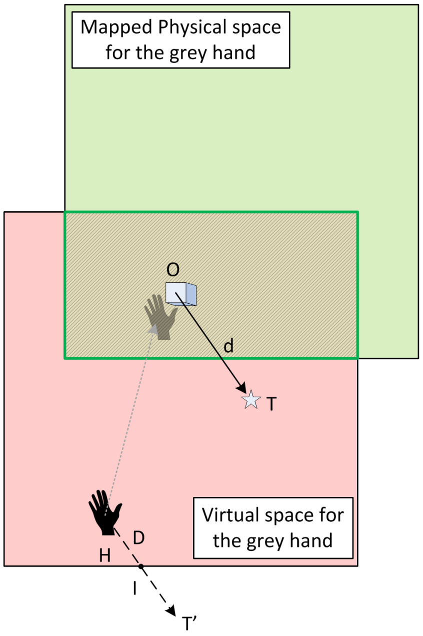
Mapped Physical space for the grey hand
O
d
T
Virtual space for the grey hand
D
H
I
T'

Fig. 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.

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 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. Go-go (Poupyrev et al., 1996) dynamically changes the CD gain to extend the virtual hand to reach the virtual object as if the user’s arm is extended to achieve a direct contact. Homer (Bowman and Hodges, 1997) also displays the virtual hand for a hand-centred manipulation technique.

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.

4.1. 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 (Fig. 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.

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

Note that D_h is the same as D in the problem definition (Fig. 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.

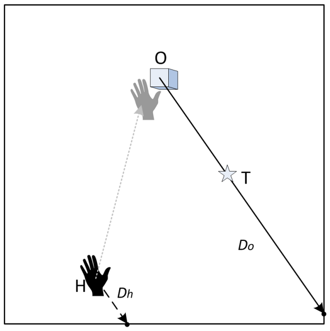

Fig. 5. Illustration of the Scaling scheme.

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 (Fig. 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.

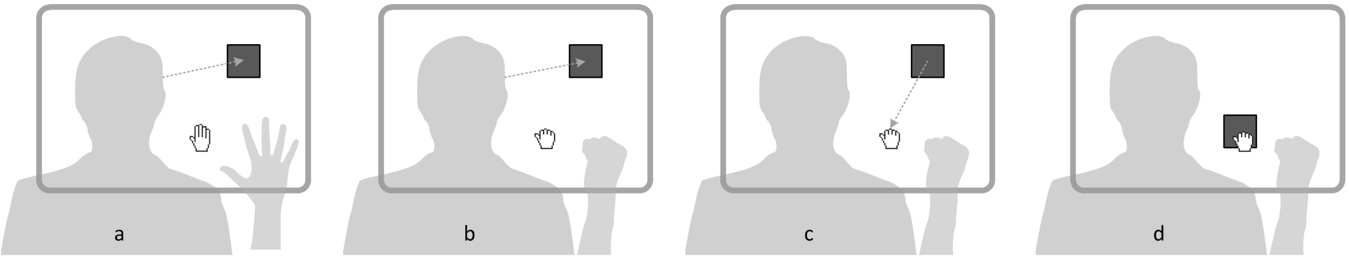

Fig. 6. Illustration of the selection phase of the Magnet method. (a) The user is looking at the object to select with hand standby. (b) He makes the grabbing gesture to confirm the selection. (c) After the confirmation, the user does not need to stare at the object anymore. The object shifts to the location of the graphic virtual hand, while the physical hand does not move. (d) The graphic virtual hand is shown grabbing the object.

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, destination location, and confirmation of drop. 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 (Fig. 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.

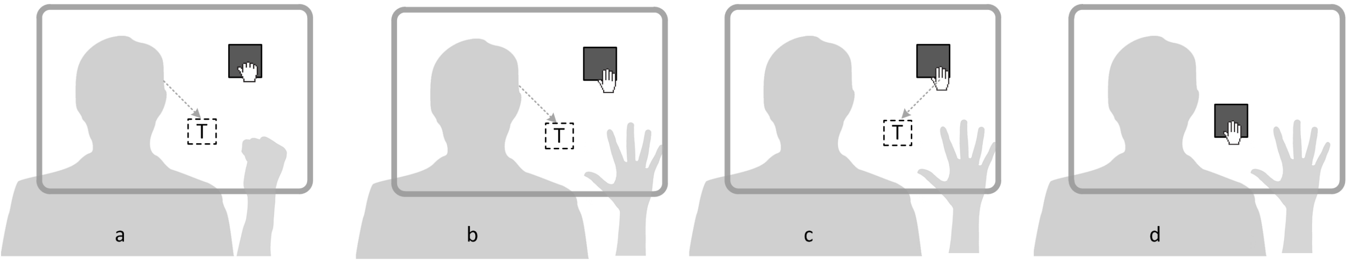

Fig. 7. Illustration of the translation phase of the Dual-gaze method. (a) After the object is selected and grabbed by the virtual hand, the user is looking at the target position. (b) The user releases their hand to confirm dropping the object to the position where they are looking at. (c) After the 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.

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.

5. EXPERIMENT

The aims of the experiment are: (1) to validate the problem we defined in Section 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 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 Fig. 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 testify 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 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-up with both conditions as the participants would not really proactively avoid the problem because they were not aware of such problem and when it would occur. The purpose here was no longer testifying the problem condition but whether it could be triggered in real interactive environments as opposed to unrealistic experimental environments in S1.

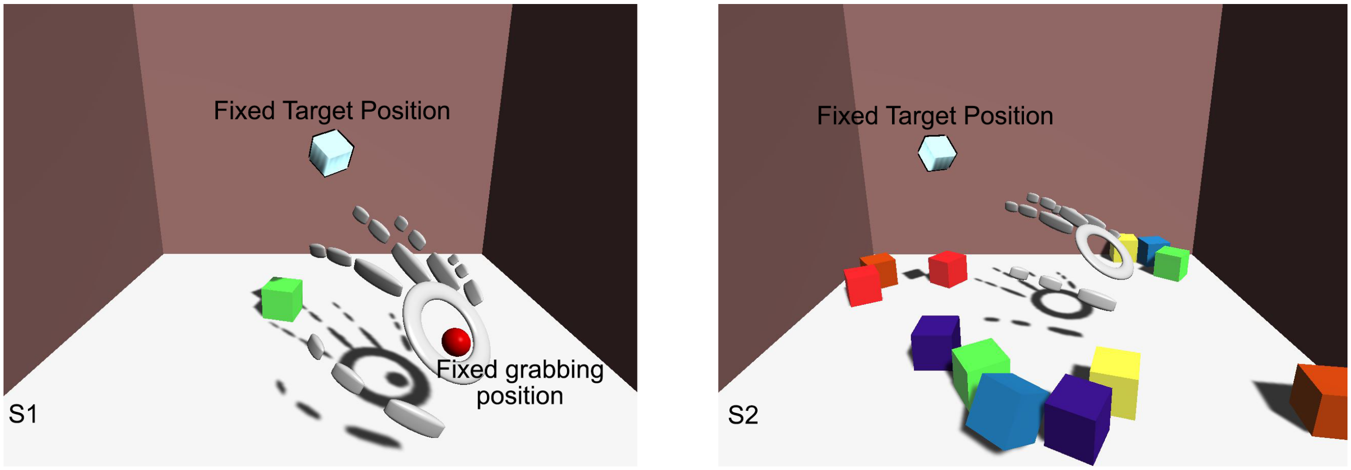

Fig. 8. Illustration of S1 and S2. In S1, the red dot indicates the fixed grabbing position. It only turns red when the virtual hand overlays with it, otherwise, it is grey. The participant can only start the trial when the dot turns red, i.e., the participant should always start to move their hand from the dot's position. In both scenarios, the highlighted floating object indicates the fixed target position where to drop the cubes.

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

5.1. Apparatus

Participants sat 66 cm 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 45 cm away from the display. The size of the virtual space was automatically generated based on the tracking space. The SDK for gesture recognition we used 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 Fig. 9.

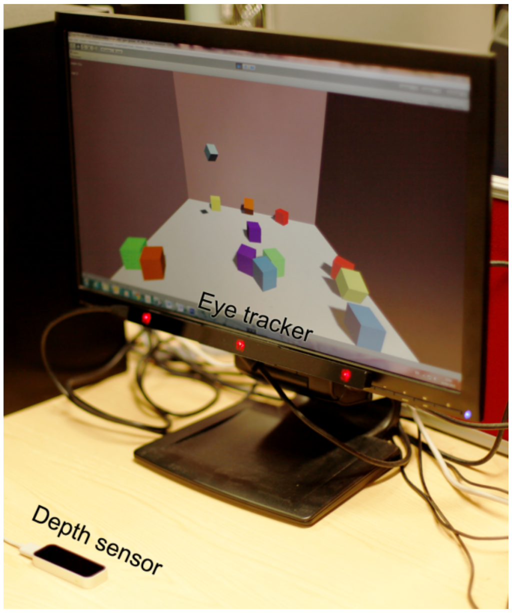

Fig. 9. Experiment setting up.

5.2. Participants

Twenty participants, 12 male and 8 female, aged between 23 and 41 (Mean ± 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).

5.3. Procedure

The user study started with a brief introduction followed by a demographic questionnaire as described in previous Section 5.2. 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 paper 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, we restricted it to be 3 degrees of freedom (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, 1996) questionnaire to fill. A post-task interview was also conducted to collect qualitative feedback.

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 feedbacks on the overall experience about what the participant liked and disliked of each tested method 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 was disappeared. For S2, the timer started when the first cube was selected and stopped when the last cube was 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.

SUS score. The SUS (Brooke, 1996) 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. 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).

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 with the others.

6.1. Completion Time

Fig. 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 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.

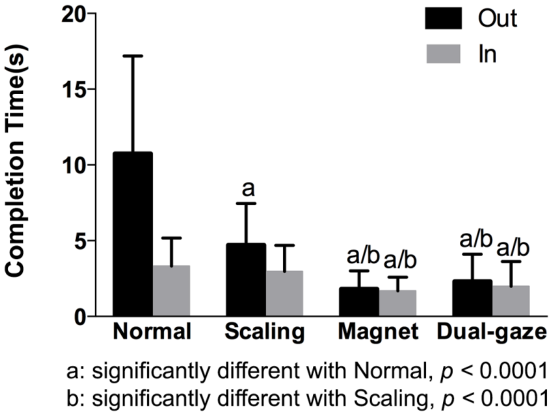

Fig. 10. Completion time for each method under the two conditions in S1. Error bar indicates the standard deviation.

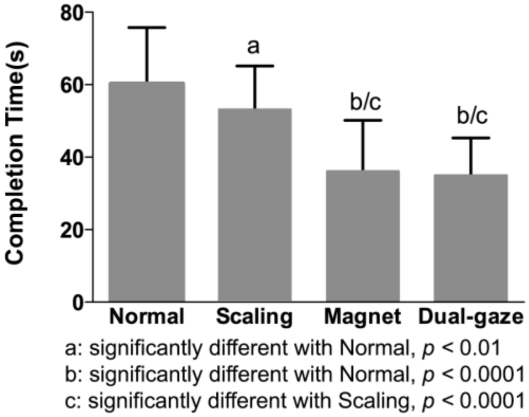

Fig. 11. Completion time for each method in S2. Error bar indicates the standard deviation.

Fig. 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 Fig. 10, that the scaling scheme shows a little improvement on 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 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% comparing to the Normal method, the Magnet method improved 39.99%, and the Dual-gaze method improved 41.80%.

6.2. Error

Table 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.

There were also errors made by the participants using the last two methods. We observed that these errors occurred due to 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.

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 1. Error rate for S1 and error count for S2

6.3. Preference

Table 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 was 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 Dual-gaze method is slightly lower than the Magnet method.

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 2. SUS score for each method

The SUS score breakdowns shown in Fig. 12 were obtained from the normalised ratings that range from 0 to 4 (the normalisation was explained in Section 5.4), so high ratings always indicate positive evaluation. The Magnet and Dual-gaze methods outperformed Normal and Scaling methods almost in all 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.

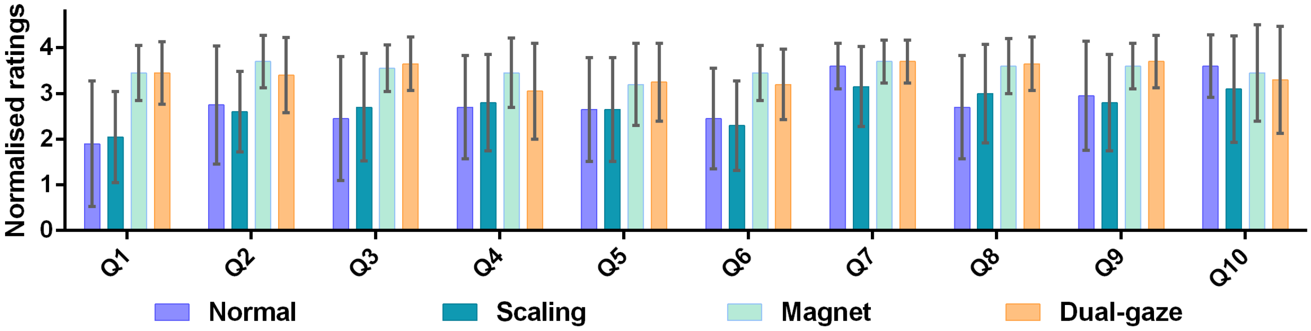

Fig. 12. SUS ratings breakdown for each method. Error bar indicates the standard deviation.

7. DISCUSSION

The results confirm that the unexpected dropping was caused by the problem we defined in Section 4. Furthermore, the proposed three methods provided circumstantial evidence that by removing some of the sufficient conditions as discussed in the background and Section 4, the problem no longer existed.

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 of the hand and arm but it does require some efforts of the eyes which may lead to eyes 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 eyes 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 our implementation, we 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.

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 would obtain preferences in continuous 3D manipulation.

8. DESIGN IMPLICATIONS

The Gaze+gesture techniques can be introduced in many applications that require fast acquisition and expressive manipulation of 3D objects to enhance the user experience. For example, it can be used for supporting interior design tasks in a virtual room setting, enriching digital LEGO building games experience for kids or enhancing the learning effects in molecular docking training. It is particularly useful with applications that are based on large and remote displays, such as smart TV and virtual reality cave. Certainly, such techniques can also provide more native support to wearable VR and AR applications. For example, eye tracking and gesture control are now available for integration with Oculus Rift. Jalaliniya et al. (2015) reported a combination of MAGIC pointing with head-mounted displays, which provided a possibility to integrate gaze modulated pointing in head-mounted VR displays. All these applications may come across the spatial misperception problem. This paper helps identify the problem and provide possible solutions along with other useful design guidance.

Not only camera-based eye tracking and depth sensors, but other devices which are absolutely mapped indirect input devices can also benefit from this study when they are dedicated to continuous manipulation. For example, the 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 constraint 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 dedicatedly 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 for the benefits that 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 tracker 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. We 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. Preferably, the eye trackers should evolve to provide higher precision and calibration accuracy but still remain cost effective. 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 Gellersen, 2015), so the gaze modulated techniques are capable of addressing the hand occlusion problem due to its indirect feature.

9. CONCLUSION

Multimodal interaction integrates advantages of each modal for a greater combined usability. Ideally, the weakness of one modal is compensated by the other, achieving a synergy as a whole. The combination of gaze and gesture can achieve a synergy of rapid and expressive interaction. In the scenarios described in this study, the gaze is capable of quick pointing but lacks natural and expressive mechanisms to support manipulation, while the gesture control can provide natural communication with an extendible gestural vocabulary of rich expressions but with rather slow pointing (Chatterjee et al., 2015).

However, the integration of multiple modalities can introduce new problems that do not exist in unimodal interactions, an obvious one is the spatial misperception problem discussed in this paper. As Norman (1999) defined, perceived affordance describes what actions the user perceives to be possible, and a real affordance describes the physical capability of a design. If we consider the tracking range as an aspect of the real affordance, the virtual space the user sees will be the perceived affordance. In many cases, vision is the only explicit perception so it is easy to consider the real affordance is consistent with the perceived affordance. In our study, that is to say, the virtual space and the tracking area are mapped consistently. When a displacement breaks the consistency, a false affordance is generated that the virtual space which looks accessible is actually inaccessible. The basic idea is to avoid the false affordance so we strive to keep the consistency between the real affordance and the perceived affordance.

This work identifies this problem and contributes to enriching the design guidance for multimodal interfaces of 3D manipulations based on eye tracker and mid-air gesture, which have great potential to be applied in many different interaction platforms. To our knowledge, we are the first to identify the spatial misperception problem, laying out the theoretical foundations for further engineering and experimentation.

A future step in our research is to refine 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. A great challenge with gesture and gaze based control is the accuracy and control fidelity so the fine grainy manipulation should be further investigated. Besides, the object distribution in the experiment 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 further test a truly random distribution where no external forces are involved as in the outer space. Moreover, Integration of further modalities such as haptic feedback is also considered in our future plan for augmenting the border perception using force feedback which can also be applied to the grab/release gestures for improving naturalness and comfortableness. Although no particular complaints about the gestures were raised during the tests. In our methods only single hand gestures were defined, one was grabbing and another was releasing the grab. Additional complex gestures and bimanual gestures can be defined to establish richer gesture vocabulary for more complicated tasks. During the implementation of these methods, we observed that participants showed preferences to certain interpolation speed as they reported them as "smooth" while some others were reported as "cumbersome". This intrigues our 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 tasks (Gielen et al., 2009).

ACKNOWLEDGEMENT

The authors would like to thank all the students and colleagues who volunteered to take part in the user study. They would like to thank Guangwei Wang for the discussions and Xun Wang for his help with the statistical analysis. Finally, they would like to thank the anonymous reviewers for their invaluable feedbacks to consolidate this work. The research leading to these results has received funding from Bournemouth University PhD scholarship –"Integration of Multiple Sensory Interactions Using Haptic and Eye-tracking Devices". The authors also acknowledge partial support from the People Programme (Marie Curie Actions) of the FP7/2007-2013/612627- "AniNex", the HEIF Fund (AANM02.X), the Chinese Postdoctoral Science Foundation (2016M600506), the FP7/2007-2013 ("Dr. Inventor", FP7-ICT-2013.8.1 611383) and the National Natural Science Foundation of China (NSFC, 51405402).

REFERENCES

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

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

Brooke, J., 1996. SUS-A quick and dirty usability scale. Usability evaluation in industry 189, 4-7.

Burdea, G., Richard, P., Coiffet, P., 1996. Multimodal virtual reality: Input-output devices, system integration, and human factors. International Journal of Human-Computer Interaction 8, 5-24.

Buxton, W., 1990. A three-state model of graphical input, Proceedings of the IFIP TC13 Third Interational Conference on Human-Computer Interaction. North-Holland Publishing Co., Cambridge, UK, pp. 449-456.

Casiez, G., Vogel, D., Balakrishnan, R., Cockburn, A., 2008. The impact of control-display gain on user performance in pointing tasks. Human–Computer Interaction 23, 215-250.

Charness, N., Holley, P., Feddon, J., Jastrzembski, T., 2004. Light pen use and practice minimize age and hand performance differences in pointing tasks. Human Factors: The Journal of the Human Factors and Ergonomics Society 46, 373-384.

Chatterjee, I., Xiao, R., Harrison, C., 2015. Gaze+Gesture: Expressive, Precise and Targeted Free-Space Interactions, Proceedings of the 2015 ACM on International Conference on Multimodal Interaction. ACM, Seattle, Washington, USA, pp. 131-138.

Cohen, P.R., Johnston, M., McGee, D., Oviatt, S., Pittman, J., Smith, I., Chen, L., Clow, J., 1997. QuickSet: multimodal interaction for distributed applications, Proceedings of the fifth ACM international conference on

Multimedia. ACM, Seattle, Washington, USA, pp. 31-40.

Fares, R., Fang, S., Komogortsev, O., 2013. Can we beat the mouse with MAGIC?, Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM, Paris, France, pp. 1387-1390.

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

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

Hales, J., Rozado, D., Mardanbegi, D., 2013. Interacting with objects in the environment by gaze and hand gestures, Proceedings of the 3rd International Workshop on Pervasive Eye Tracking and Mobile Eye-Based Interaction, pp. 1 -9.

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

Jalaliniya, S., Mardanbegi, D., Pederson, T., 2015. MAGIC pointing for eyewear computers, Proceedings of the 2015 ACM International Symposium on Wearable Computers. ACM, pp. 155-158.

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

Marin, G., Dominio, F., Zanuttigh, P., 2015. Hand gesture recognition with jointly calibrated Leap Motion and depth sensor. Multimedia Tools and Applications, 1-25.

McLaughlin, A.C., Rogers, W.A., 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, 1-15.

Mylonas, G.P., Kwok, K.-W., James, D.R.C., Leff, D., Orihuela-Espina, F., Darzi, A., Yang, G.-Z., 2012. Gaze-Contingent Motor Channelling, haptic constraints and associated cognitive demand for robotic MIS. Medical Image Analysis 16, 612-631.

Norman, D.A., 1999. Affordance, conventions, and design. Interactions 6, 38-43.

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

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

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

Quek, F., McNeill, D., Bryll, R., Duncan, S., Ma, X.-F., Kirbas, C., McCullough, K.E., Ansari, R., 2002. Multimodal human discourse: gesture and speech. ACM Trans. Comput.-Hum. Interact. 9, 171-193.

Sauro, J., 2013. 10 Things To Know About The System Usability Scale (SUS), http://www.measuringu.com/blog/10-things-SUS.php.

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

Simeone, A.L., Bulling, A., Alexander, J., Gellersen, H., 2016. Three-Point Interaction: Combining Bi-manual Direct Touch with Gaze, Proceedings of the International Working Conference on Advanced Visual Interfaces. ACM, Bari, Italy, pp. 168-175.

Simeone, A.L., Gellersen, H., 2015. Comparing indirect and direct touch in a stereoscopic interaction task, 2015 IEEE Symposium on 3D User Interfaces (3DUI), pp. 105-108.

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

Song, J., Cho, S., Baek, S.-Y., Lee, K., Bang, H., 2014. GaFinC: Gaze and Finger Control interface for 3D model manipulation in CAD application. Computer-Aided Design 46, 239-245.

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

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

Turner, J., Alexander, J., Bulling, A., Schmidt, D., Gellersen, H., 2013. Eye pull, eye push: Moving objects between large screens and personal devices with gaze and touch, Human-Computer Interaction–INTERACT 2013. Springer, pp. 170-186.

Velloso, E., Turner, J., Alexander, J., Bulling, A., Gellersen, H., 2015. An Empirical Investigation of Gaze Selection in Mid-Air Gestural 3D Manipulation, Proc. of the 15th IFIP TC13 Conference on Human-Computer Interaction (INTERACT 2015). Springer International Publishing, pp. 315-330.

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

Zhai, S., Morimoto, C., Ihde, S., 1999. Manual and gaze input cascaded (MAGIC) pointing, Proceedings of the SIGCHI conference on Human factors in computing systems. ACM, Pittsburgh, Pennsylvania, pp. 246-253.

Zhang, Y., Stellmach, S., Sellen, A., 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., 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, Cham, pp. 570-577.