

A Study of Selection and Navigation Techniques Using Kinect in VR

Peter Dam, Manuel Loaiza, Luciano Soares, Alberto B. Raposo

*Tecgraf – Computer Graphics Group
PUC-Rio (Pontifícia Universidade Católica do Rio de Janeiro)
Rio de Janeiro, Brasil
{peter, manuel, lpsoares, abraoso}@tecgraf.puc-rio.br*

ABSTRACT

This paper proposes interaction techniques to use Microsoft Kinect in Virtual Reality (VR) applications. Most of the proposed techniques are used in a similar fashion with devices other than Microsoft Kinect, that will allow a user to select objects in a virtual scene and navigate throughout that scene. Kinect was chosen because it is low cost, easily portable and does not require the user to have any equipment attached or held by him, increasing the user's sense of immersion. Techniques are discussed, pointing out advantages and disadvantages of each one, with the intention of creating a knowledge base for the development of practical VR applications.

Keywords: Microsoft Kinect; hands-free interaction; selection; navigation; 3D user interaction; immersive environments.

1 INTRODUCTION

Currently we have several devices available for use in an interactive virtual environment, but few solutions are capable of allowing multiple kinds of interaction with the scenario. One of the problems of immersive environments is the navigation throughout the scene. An apparently natural idea that immediately comes to mind to allow the user to move without using traditional handheld devices is an omnidirectional treadmill [5], but it is extremely expensive to build and isn't portable, requiring a special room to set up. Recently, however, due to the gaming industry, technology has evolved with the creation of a few natural user interface devices such as Nintendo WiiMote, Sony Move and Microsoft Kinect.

In the latter we realized the potential to develop a study that will pinpoint techniques allowing the user to interact and navigate in a virtual environment without use of handheld devices, in hopes of increasing the user's immersion and experience. Also, due to not needing to hold anything, this opens the opportunity to use finger tracking, increasing the power the user has over the environment.

Bowman and Hodges [2] describe that most of the interactions in a virtual environment fall into one of three categories: selection, manipulation and navigation. Taking into account these categories, we focused on two of them: selection and navigation. We believe these two categories are most important to be able to interact with a scene satisfactorily. For each category a few techniques were encountered or proposed using Microsoft Kinect, inspired on similar techniques used with different devices. This paper studies the apparent advantages and disadvantages of each of these techniques.

In section 2 we present related work that, in some way, inspired the techniques proposed in this paper. Following that we will see, in section 3, the proposed selection and navigation techniques, where we show advantages and disadvantages of each of these techniques.

2 RELATED WORK

In the last two years, academic and commercial devices based on innovative technologies have been proposed in order to increase the usability and experience of natural interaction that a user feels when immersed in a game or virtual scenery.

We analyzed some of these devices and did a survey of interaction techniques that have been proposed along with them. We summarized the main features of interaction that these techniques offer and described them below.

2.1 Microsoft Kinect Games

A list of Microsoft Kinect games was tested in an attempt to identify selection and navigation techniques. In these games very few navigation techniques were encountered, basically only one idea was present, used in the game Kinect Sports [10], which consists in the user "walking in place" to move the avatar forward. The avatar moves in a fixed direction and cannot turn or walk sideways nor backwards, so although it could be considered a technique it is extremely limited. As for selection techniques there were two that were used throughout all the tested games. One is hovering the virtual hand (controlled by the user's physical hand) over an item (usually a UI option) and keeping it relatively still for a short period of time. This is found in games such as Kinect Sports and Kinect Adventures [9]. The other technique, found in Dance Central [3] tracks the right hand on the vertical axis to move between menu options and a swipe of the hand from right to left to confirm selection, and left to right to cancel or move back in the menu hierarchy. Although these techniques were identified, none of them were used in the gameplay itself, restricted only to 2D menus.

2.2 Non-Kinect Related Work

Beside the Microsoft Kinect games, some other systems not using Kinect serve as inspiration for several techniques presented here.

2.2.1 v-Glove

Rodrigues et al. [11] developed a wireless glove tracked by wii mote controllers, feeding a system with the real-world position of the user's hand. In his work he proposes the use of a "virtual touch screen" to detect the intention of interaction.

Tosas and Li [12] proposed a mixed reality environment that allows users to interact with a virtual touch screen. The idea is that you can interact with elements on a graphical user interface with "clicks" made using your bare hands.

Rodrigues et al. [11] used this idea defining a virtual screen in real-world space. Once the user's hand reaches this screen a "click" is detected at the X and Y position where the hand crossed the virtual screen.

2.2.2. Dance Pad

The dance pad, originating from the Dance Dance Revolution [4] game, consists of a metallic pad placed on the floor with arrows pointing in four directions: forward, backward, left and right. In addition to those, some may have diagonal arrows at 45° as well.

With the pad oriented towards the display Beckhaus, Blom and Haringer [1] mapped the four directional arrows to positional transformation in a straight-forward fashion, while using two directional arrows (forward-left and forward-right) to control camera rotation.

2.2.3. Touch-screen controls

In some games for touch-screen devices (such as iPhone, Android touch-screen phones and iPad), more commonly in FPS (first-person shooters) such as DOOM Classic [6], directional pads are placed in both bottom corners of the screen. These pads are activated by touch, usually the thumb due to the anatomic nature of the device. The left pad controls character movement in the scene while the right pad controls the camera's rotation.

3 DISCUSSION

Using the Microsoft Kinect several techniques were implemented for further studies in virtual reality environments. These techniques focused on selection and navigation through a 3D environment.

3.1 Selection Techniques

3.1.1. Hover

Inspired by the majority of Microsoft Kinect games, this technique consists on the use placing a virtual hand over an object. After a short period a timer will appear indicating that an object will be selected (Figure 1). The user should keep the virtual hand (controlled by his physical hand) hovering over the object in case he desires to select it. Once the time runs out the object will be selected. This technique is similar to that encountered in eye tracking, as seen in Hansen, Andersen and Roed [7].



Figure 1: Hover selection technique

Advantages – This method is the most precise amongst the methods studied because the user's hand does not need to move along the Z axis (towards or away from the screen).

Disadvantages – Sometimes the timer appears even though the user did not have the intention of selecting the object, similar to what Jacob [8] calls the "Midas Touch", which consists in the user sending commands by accident, losing control of the

application. It's necessary to find a comfortable delay from the

moment the virtual hand hovers an object until the appearance of the counter. Another problem is the difficulty to unselect an object. Since basically this technique was conceived for 2D menus, no means of unselecting was encountered.

Discussion – The *hover* method seems to work best for "instant selection", which is when there is no persistence of the selection state: pushing a button, opening a door or menu options, for example. As a selection method that intends to lead to manipulation, this technique does not appear to be the most adequate.

3.1.2. Push

Inspired by v-Glove [11], this technique requires the user to stretch his arm towards the screen. Once selected an object maintains its selected state until the user stretches his arm towards the screen again.

Advantages – There is no visual pollution with timers. The selected objects are easy to manipulate because it's uncommon for the user to accidentally unselect an object.

Disadvantages – Sometimes the user may get confused by attempting to interact with a second object before unselecting the first since the gesture is the same. Another problem is the precision when attempting to select an object because the user must move his hand along the Z axis and, during this movement, he might unintentionally move it along X or Y axis as well, potentially losing the focus upon the desired object and consequentially failing to select it, as noted by Vogel and Balakrishnan [13].

Discussion – This method works best to select objects that will have a prolonged manipulation, such as carrying an object from one location to another. It may be interesting to study a way to replicate the haptic feedback present in v-Glove, according to Wang and MacKenzie [14] the user's performance decreases in these sorts of actions when there is no feedback.

3.1.3. Hold

Similar to "*Push*" technique, the user places the virtual hand over an object and stretches his arm towards the screen. Once his arm is stretched the selection takes place, but this time unselecting occurs when the user withdraws his arm.

Advantages – The advantages are the same as "*push*", but furthermore there is no confusion with unselecting, since the selecting and unselecting gestures are complementary.

Disadvantages – This method is physically more tiresome since the user needs to keep his arm stretched during manipulation. As in "*Push*" there is a problem with precision.

Discussion – This technique works well for quick interactions such as pulling a lever. It seems more natural as well, because it appears to be the closest to how a person would interact with an object in the real world, but is very tiresome for prolonged interactions. Like with "*Push*" it would be interesting to provide some sort of haptic feedback to the user.

3.2 Navigation Techniques

3.2.1. Virtual Foot DPad + Body Turn

Beckhaus, Blom and Haringer [1] present a navigation method based on the "*dance pad*", a physical device that has directional arrows activated by stepping on them. Similarly the technique we propose here works like a directional controller activated by the feet, however the buttons are virtual. Both feet work to activate the buttons and the system calculates the direction based on the direction of the user's feet to his waist. Y axis (height) is not taken into account in this calculation (Figure 2). Unlike the work of Beckhaus and colleagues, where the camera is controlled by

directional arrows, this technique requires the use to turn his shoulders in either direction and the camera will turn in that direction. Furthermore if the user tilts his torso backwards or forward the camera will turn upwards or downwards, respectively.

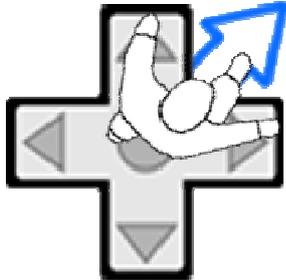


Figure 2: Virtual Foot DPad

Advantages – The user does not need to move around the physical environment, only needing to move his foot towards the desired movement direction. The user can, if he wants, move around the physical environment, since the technique is based on distance between certain body parts.

Disadvantages – Moving backwards is complicated because sometimes the system will interpret as forward movement when one foot is in back. This happens because when the user places one foot back he ends up leaving the other foot forwards. Also despite the user being able to move around the physical environment, the technique will detect some commands by accident due to the gestures used being similar to the natural human walk.

Discussion – Although susceptible to errors, this method allows the user to freely move around the physical environment, which is highly desirable in immersive environments such as a CAVE.

3.2.2. Dial DPads

Present in some multi-touch games (such as games for the iPhone), this technique places directional pads in each bottom corner of the screen. When the user places his hand over a certain direction on one of these pads while keeping his arm stretched he may control the avatar's position and the camera's rotation (there is one dial for each action) – Figure 3.



Figure 3: Dial DPads

Advantages – This technique does not depend on feet tracking, potentially allowing the user to be seated, for example. This technique also allows the user to move freely around the physical environment.

Disadvantages – Interaction with objects could be affected in case the user wants to interact with an object that is in one of the bottom corners of the screen. It is also very difficult to move the avatar and control the camera at the same time, since the user would have to coordinate one hand for each task simultaneously. It could also be hard to control the camera or move the avatar while manipulating an object for the same reason.

Discussion – Because this method does not require the user to move his feet this method could be interesting in certain situations.

3.2.3. Virtual Circle + Body Turn

Based on the user's physical position during calibration a virtual circle is created around that spot. The user then walks in any direction and, once he exits that virtual circle a vector is created between his current position and the center of the circle. That vector indicates the desired movement (Figure 4). To stop moving the user needs to step back into the circle. As with "Virtual Foot DPad" for camera control the user turns his shoulders or tilts his torso towards the desired looking direction.

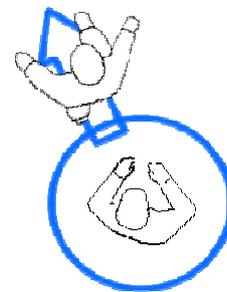


Figure 4: Virtual Circle

Advantages – Navigation is easy because the user simply walks in the direction he wants the avatar to move. Also he can control the movement speed by moving further from the circle.

Disadvantages – The user may lose reference of where the virtual circle is, making it difficult to stop the avatar's movement. It could also be tiresome having to move every time the user wants to move the avatar. Also the user cannot freely move around the physical environment, since that would make the avatar move as well.

Discussion – This technique is the most precise and easiest to use according to preliminary tests. The inability to freely move around the physical environment could be prohibitive, however, so it would be interesting to study a manner of allowing the user to move freely, as well as a way to allow the user to reset the location of the virtual circle.

4 CONCLUSION AND FUTURE WORK

This study shows several selection and navigation techniques for virtual environments using only Microsoft Kinect. We analyzed the potential uses of each proposed technique, as well as its advantages and disadvantages. Although we haven't conducted formal user studies, our initial experiments with the proposed techniques indicate that each technique presents singular characteristics that make it more appropriate for specific tasks. The idea of this paper is to create an initial survey upon which we may organize and categorize a knowledge base for interaction techniques for 3DUI using Microsoft Kinect.

As future work, besides formal user tests, we plan to include manipulation techniques. We also plan to study the combination of navigation, selection and manipulation techniques and to

improve the techniques based on the users' experiences, since that through observation we can discover new potential ways to interact using Microsoft Kinect.

It is also interesting to study in depth the differences in applicability of the proposed techniques in immersive and non-immersive environment.

REFERENCES

- [1] Beckhaus, S., Blom, K.J., Haringer, M.; Intuitive, Hands-free Travel Interfaces for Virtual Environments; *New Directions in 3D User Interfaces Workshop of IEEE VR 2005 (2005)*
- [2] Bowman, D.A., Hodges, L.F.; Formalizing the Design, Evaluation, and Application of Interaction Techniques for Immersive Virtual Environments; *Journal of Visual Languages and Computing (1999)*
- [3] Dance Central, *Microsoft*; available at <http://marketplace.xbox.com/pt-BR/Product/Dance-Central/66acd000-77fe-1000-9115-d802545607d3>
- [4] Dance Dance Revolution, *DDR Game*; available at <http://www.ddrgame.com/>
- [5] Darken, R., Cockayne, W., Carmein, D., The Omni-directional Treadmill: A Locomotion Device for Virtual Worlds, Proceedings of UIST'97, (1997).
- [6] DOOM Classic for iPhone, *id Software*; available at <http://itunes.apple.com/us/app/doom-classic/id336347946?mt=8>
- [7] Hansen, J.P., Andersen, A.W., Roed, P.; Eye-Gaze Control of Multimedia Systems; *ACM Symposium on Eye Tracking Research & Applications (1995)*
- [8] Jacob, R.J.K., et al.; Interaction styles and input/output devices; *Behaviour & Information Technology (1993)*
- [9] Kinect Adventures, *Microsoft*; available at <http://marketplace.xbox.com/pt-BR/Product/Kinect-Adventures/66acd000-77fe-1000-9115-d8024d5308ed>
- [10] Kinect Sports, *Microsoft*; available at <http://marketplace.xbox.com/pt-BR/Product/Kinect-Sports/66acd000-77fe-1000-9115-d8024d5308c9?nosplash=1>
- [11] Rodrigues, P. G.; Raposo, A. B.; Soares, L. P., A Virtual Touch Interaction Device for Immersive Applications. *International Journal of Virtual Reality*, v. 10, n. 4, p. 1-10, Dec. 2011.
- [12] Tosas, M., Li, B., et al., Virtual Touch Screen for Mixed Reality. *ECCV 2004 Workshop on HCI*, Prague, Czech Republic, May 2004
- [13] Vogel, D., Balakrishnan, R.; Distant Freehand Pointing and Clicking on Very Large, High Resolution Displays; *ACM (2005)*
- [14] Wang, Y., MacKenzie, C.; The role of contextual haptic and visual constraints on object manipulation in virtual environments; *ACM CHI Conference (2000)*