Periphery Triggered Menus for Head Mounted Menu Interface Interactions

Peter Mitchell
Faculty of Science and Engineering
Flinders University, Adelaide, South Australia
Peter.Mitchell@flinders.edu.au

Brett Wilkinson
Faculty of Science and Engineering
Flinders University, Adelaide, South Australia
Brett.Wilkinson@flinders.edu.au

ABSTRACT
Interaction with head mounted displays is predominantly through use of additional controllers or the keyboard to initiate pointing, selections, and navigation. However, most modern head mounted displays have an orientation sensor to determine how the user's view should be displayed. There is the potential to use this sensor as an input mechanism. Our research explores a method of interaction using just head movements called Periphery Menus. These are menus triggered by looking quickly in predetermined directions to reveal a contextual menu from the periphery of a user's vision. We discuss a pilot study of this interaction technique, using a serious games approach to experimental testing. Experiment results indicated the interaction provided a successful, engaging, and repeatable experience.

Author Keywords
Head Mounted Display, User Interface, Virtual Reality, Serious Games

ACM Classification Keywords
H.5.2. Input devices and strategies, Graphical user interfaces

INTRODUCTION
The availability of access to Virtual Reality (VR) has been rising due to the release of consumer Head Mounted Displays (HMDs) such as the: Oculus Rift, HTC Vive, and PlayStation VR. The cheaper pricing of higher quality displays containing head tracking has provided both developers and consumers with an opportunity to experience VR with a larger sense of immersion. This extends to Augmented Reality (AR) as well, either in the case of the Microsoft HoloLens or additional rigging of hardware to VR HMDs (Höll et al. 2016). This paper will focus on the VR implementation side of HMD interface interactions.

OzCHI '16, November 29-December 02, 2016, Launceston, TAS, Australia
 ISBN 978-1-4503-4618-4/16/11…$15.00
DOI: http://dx.doi.org/10.1145/3010915.3010964

There are a wide variety of different interaction methods that have been researched or implemented with application to HMDs. The HTC Vive for example has controllers allowing for motion tracking and selection inputs. Google's Tilt Brush (Google, 2016) application demonstrates how an interface can be attached to the HTC Vive controller for the purpose of digital 3D art. Handheld controller free interaction methods can use devices such as the LEAP Motion Sensor or Microsoft Kinect to track parts of the body. Hardware can be used in combination to correlate a wider array of gestures, such as the combination of both the LEAP and Microsoft Kinect at the same time (Marin et al. 2014). Other body parts such as the face have also been explored as a point of interaction input, using the face as a gesture detection surface (Serrano et al. 2014), or nodding and tilting of the head as a gesture input (Špakov et al. 2012).

Interfaces for HMDs are presented as surfaces or elements of the virtual environment, such as in Tilt Brush (Google, 2016), this is to reduce interface clutter that a standard WIMP type interface presents (van Dam, 1997). Interfaces can become immersive as a tool for interaction when represented by physical objects or body parts to provide a natural user interface (Bowman and Wingrave, 2001).

The problem with many of the methods of interface interaction at the moment is the diversity in the number of physical tools. When considering the market of tools from a consumer perspective this can be overwhelming. This also then limits the ability for developers to choose to support many of the specific hardware interaction solutions unless they are included with leading HMDs. With the assumption that all HMDs currently available have some form of orientation tracking, this allows for research into head gesture tracking as a means of menu interaction.

The Periphery Menu System, discussed in this paper, provides an interaction method that can be used to reveal and interact with hidden menus. A pilot experiment has been conducted to test the usability of the Periphery Menu System. The experiment design and preliminary results for this experiment are discussed below. The participants of the experiment found the interaction method worked for the completed tasks and indicated on average they would use the system in the future. It is hoped that this menu interaction method will be found useful for those seeking a way to present alternative methods of interface interaction.

PERIPHERY MENU SYSTEM

The Periphery Menu System is intended to provide access to hidden or less frequently required functionality. Menus that are revealed using the head as an interaction tool could be used throughout an application. In this experiment the menus are ones that you would access frequently for a specific use case: a tower defence game. All interactions within this experiment use a “look at” type interaction to perform selections.

Revealing the Menu

Revealing of the Periphery Menu System is completed by a rapid gesture on either the X or Z axis (left/right/up/down). Behind the scenes a threshold value for the primary axis is set. Data for each interaction is only tracked up to a maximum interaction time (0.25s to 0.35s for this experiment). The maximum difference between the values on each axis calculated. If the rotation on the X or Z axis is over the primary threshold, then the other value and the Y axis maximums are compared against corresponding thresholds. If these are lower than the thresholds, the direction of interaction is determined using the first and last interaction events in the time window. If there is an appropriate menu corresponding to the detected interaction direction it will be displayed. The location of the menu once triggered is offset at either A or B in Figure 1 (left/right) depending on the side triggered. This is to prevent immediate interaction with the menu. Due to the menu system relying purely on the orientation sensor of a head mounted display, this approach should be applicable for most modern head mounted displays currently available at the time of writing. Figure 1 shows an example of rotating the FOV by 25 degrees and then from the arrow direction a menu would appear at position B from a right gesture.

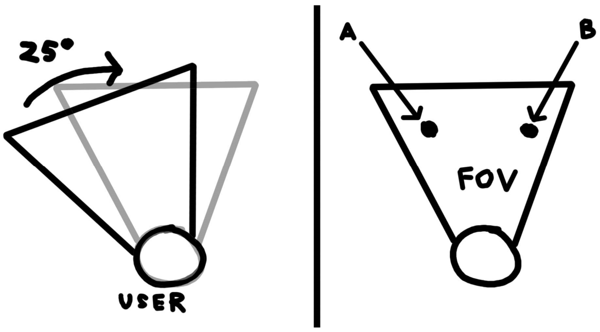

Figure 1. Revealing the Menu (Top Down FOV Example)

Visual Menu Design

The menu design used for this experiment was intentionally simplistic. It is certainly possible for the menu design to take any appearance with as much complexity as a developer wishes. For this experiment the interface consisted of a menu title, and four menu buttons. Importantly the text sizing of the menu was made large to account for the lower DPI of the Oculus DK2 (Developer Kit 2). User’s were provided a simple crosshair as a reference point for their look-at target. Selection was executed with a dwell mechanic. This crosshair would appear as orange by default, and then fade into green as selections were progressing. To ensure there wasn’t excessive menu generation an enforced delay between each menu presentation was used.

EXPERIMENT DESIGN

Participants

A pilot run of the proposed interaction method was conducted at Flinders University. There were 23 participants who completed the full experiment. Most participants came from a computing or engineering background. All participants were students studying at the university. Recruitment used a narrative trailer to provide an enticing experience for participants to engage with. This helped to push the use of a serious games approach that was used in this experiment. No prior knowledge was required, although there was the general assumption that the pool of participants being sourced from a university in computing and engineering fields would have experience with computers and user interfaces. The Oculus Rift DK2 with SDK v0.8 was used for this experiment with the software having been developed in Unity 5.3.2f1.

EXPERIMENT OVERVIEW

The experiment consisted of the following stages that each participant would complete:

- A pre-experiment questionnaire.
- A calibration data collection task.
- A tower construction task.
- An interactive tower defence game task.
- A post-experiment questionnaire.

During each of the three middle stages introductory text and imagery was presented to the user within the HMD. This explained the goal of what they needed to do to complete each puzzle. Additionally, after each of these stages a single ease question (SEQ) was asked regarding the usability of the functionality within the constraints of that puzzle.

Calibration Data Collection Task

This task was designed to explore minor variation between the three different sensitivity configuration thresholds for triggering the periphery menu system. A low, medium, and high value was selected. For all of these tasks, triggers were only used for looking left and right (the X rotation axis).

Sensitivity	X	Z	Y	Time (s)
Low	25	30	30	0.35
Medium	25	30	30	0.3
High	25	25	25	0.25

Table 2. Sensitivity Values for Periphery Menu Triggering

In Table 1 it can be seen that the main difference separating all three is the Time variable (shown in seconds). This is the maximum amount of time over which the interaction detection can be made. X, Z, and Y are all euler angles. The X is a threshold value that must be reached, and if the Z or Y thresholds are passed the entire interaction is ignored. These were set to be very high so that although the menu would trigger more often, the data could be collected relating to how high these values were in realistic situations.

To complete this task, participants were asked to work through 24 steps in a random order. This consisted of completing each of the 12 task variations twice. For each sensitivity the user would have to complete an action of revealing the periphery menu, or look far enough to a specific location as seen in Figure 2. This had to be done for menus to both left and right. 24 combinations = 2 * (3 sensitivities * 2 directions * (reveal menu + no reveal menu)). Figure 2 shows the panel at the far left on the left pane. A matching one was on the far right. And an instruction dialog was centred to the user with the current task displayed on a button at the bottom of the interface.

Figure 2. Calibration Data Collection Task

Tower Construction Task

The tower construction task allowed the participants to experience a task with no time sensitive constraint. Four different types of towers (basic, frost, swarm, and explosive) were presented which the participants could customise. They could: increase the damage, increase the range, or decrease the time between shots. Six modifiers had to be chosen for each tower, which could consist of any variation of the modifiers. These were selected through the periphery menu (from looking to either the left or right), with an additional option to remove the previous modifier. The visualisation of these modifiers was a simple block that could then be stacked with other blocks. An example screenshot of this task can be seen below in Figure 3. Medium sensitivity from Table 1 was used for this task and the final task.

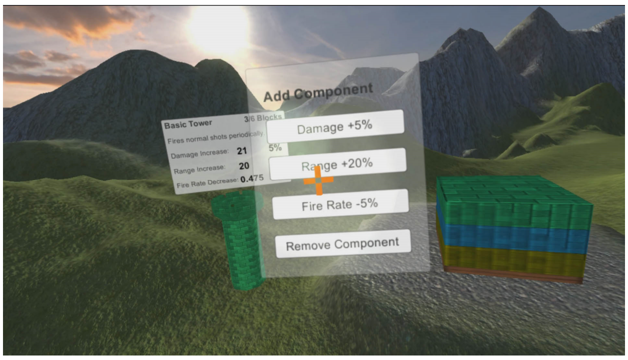

Figure 3. Tower Construction Task

Tower Defence Task

This task was the primary drawing point to attract participants. During this task the participant would use the customised towers from the previous construction task to deploy towers against waves of enemies. In this case the enemy units took the form of spiders with varying speeds, health values, appearances, and some with the ability to attack the player's towers. The player engaged with this activity by building towers through selection of a preferred tower to build using the periphery menu triggered by looking to the right. This was restricted by a small amount of currency given at the start, after each round, and awarded from destroying enemy units. When players had a tower selected they would be able to trigger a periphery menu to the left. This menu provided options to: repair (with relative cost to how badly damaged a tower was), destroy (refunding a partial amount of the cost to build), move (only in downtime between enemy waves), and deselect. Each downtime consisted of a 60 second period. One to start with, then after each of four separate sets of enemies. This task allowed participants to experience the interaction method within a time sensitive context. The time sensitive element came largely from having to proactively build new towers while maintaining repairs on the existing ones. Figure 4 shows an example of what would be seen while interacting with the tower defence game.

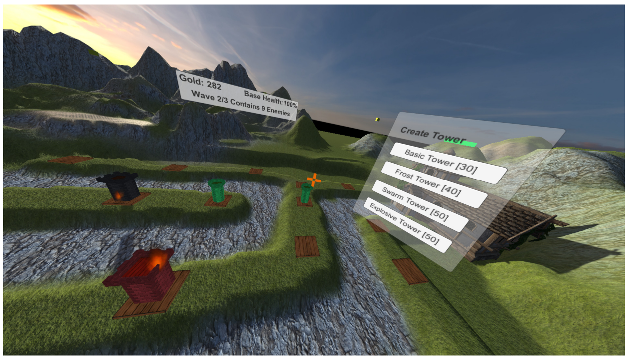

Figure 4. Tower Defence Task: Create Tower Menu

RESULTS

In general, the response from participants was positive. Figure 5 shows some of the general metrics indicating how well the experience was overall. On a 10-point scale the mean average response in regard to whether the gesture was found to be useful was 6.7 with a σ of ±2. More significant was the indication that participants would be interested in using the system in future with a mean of 7.5 and a σ of ±2. On average users found the system displayed at the correct times when it was intended more often than when they didn't mean it to display. With a mean average of 7.9 (σ of ±1.3) for correct times, and 5.4 (σ of ±2.1) for wrong times. This high number of wrong times can be accounted for by the high threshold values (for the z and y axis) to collect a larger set of error cases for future analysis.

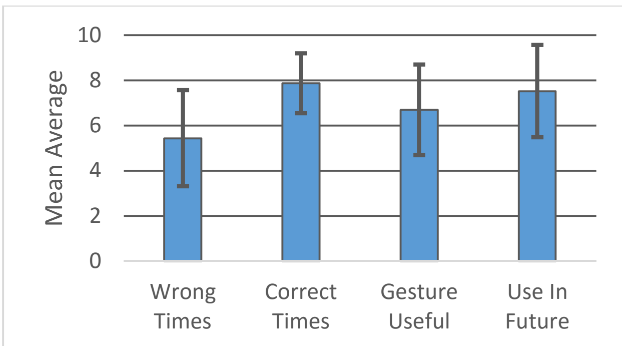

Figure 5. User Interaction Overall Metrics

After each of the tasks, the participants were asked a SEQ to evaluate how they would rate the usability of functionality. As these were asked from within the application with a user interface the ratings were ranked from 1 to 5 to simplify input. Three separate questions were asked after the calibration task to evaluate the participant's perspective on the usability with each of the low, medium, and high sensitivity settings. After the construction and tower defence tasks participants were asked about the periphery menu usability in regard to the untimed (construction), and timed (tower defence) tasks. The results for these questions can be seen in Figure 6. To summarise these results: low sensitivity had a mean of 3.4 (σ of ±0.2), medium sensitivity had a mean of 3.7 (σ of ±0.3), high sensitivity had a mean of 3.6 (σ of ±0.6), untimed construction activities had a mean of 4.1 (σ of ±0.4), and the timed tower defense had a mean of 3.6 (σ of ±0.6).

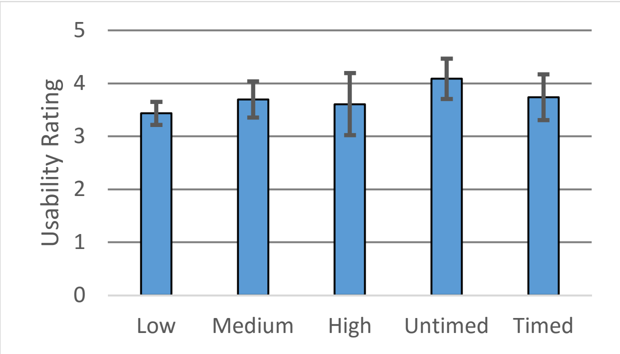

Figure 6. Feature Usability Mean Averages

What these results indicate is that the interaction experience of the participants using the defined sensitivities was roughly similar, with the medium sensitivity being the preferred choice more consistently. The high sensitivity score was close in mean to the medium, but had a higher variance. The untimed activity was preferred to the timed activity. The small preference toward untimed could be explained by the increased difficulty in rapidly accessing the menu system while attempting to play a game requiring many menu interactions.

When asked about the features they enjoyed participants responded saying that they found the system: easy to use, good for hiding complexity, and improved immersion through the hands-free nature. Some of the difficulties participants identified included: accidental revealing of the menu, sometimes the menu got in the way with no easy way to close it, and some minor fatigue from use of neck for interactions.

CONCLUSIONS

This paper has explored the use of a Periphery Menu System triggered with a rapid gesture of the head to the left or right. This system is intended to provide a menu that can be used by most of the head mounted display models currently available at the time of writing. It accomplishes this by only using the orientation sensor to control the interaction. It is anticipated that the introduction of an independent selection device (eg, game console controller) would improve the interaction experience, but this system is designed to allow the hidden menu to exist separately if desired. Through the described experiment and accompanying results, it can be seen that there is a positive reaction to the usability of the menu system. Participants of the experiment indicated that they would likely use the system themselves in the future. The main issue that was highlighted during the experiment was the higher than desired frequency of the menu appearing when not intended. This was largely due to the higher thresholds to allow for more aggressive data collection. This along with any other issues that were acknowledged by participants will be addressed in future experiments.

REFERENCES

Bowman, D.A., and Wingrave, C.A. 2001, Design and evaluation of menu systems for immersive virtual environments, In Proc. IEEE'2001 Virtual Reality.

Holl, M., Heran, N., and Lepetit, V. 2016, Augmented Reality Oculus Rift: Development Of An Augmented Reality Stereoscopic-Render-Engine To Extend Reality With 3D Holograms, Inst. for Computer Graphics and Vision Graz University of Technology, Austria, URL: http://arxiv.org/pdf/1604.08848v1.pdf.

Google 2016, Tilt Brush, URL: https://www.tiltbrush.com/

Marin, G., Dominio, F., and Zanuttigh, P. 2014, Hand gesture recognition with leap motion and kinect devices, In Proc. of ICIP'2014, IEEE.

Špakov, O., and Majaranta, P. 2012, Enhanced gaze interaction using simple head gestures, In Proc. of UbiComp'12, 705-710.

Serrano, M., Ens, B., and Irani, P. 2014, Exploring the Use of Hand-To-Face Input for Interacting with Head-Worm Displays, In Proc. of SIGCHI'14, 3181-3190.

van Dam, A. 1997, Post-WIMP user interfaces, ACM 1997, 63-67.