Pointing Gestures and Object Manipulation Gestures Using a Hand Tracking System and a Head Mounted Display

Kai Groetenhardt
Herrngartenstraße 16
65185 Wiesbaden
+49 151 61424734
kai.b.groetenhardt@student.hs-rm.de

University of Applied Sciences
Wiesbaden Rüsselsheim
Kurt-Schumacher-Ring 18
65197 Wiesbaden

Abstract
To interact within virtual reality environments, input devices like mouse, keyboard, game controllers, joysticks, data gloves or optical tracking systems are used. But using a Head Mounted Display (HMD) could make it difficult to use input devices like the combination of mouse and keyboard because the user cannot see them. Potentially, it is easier to use input devices that do not need to be seen like the hand and finger tracking device Leap Motion Controller (LMC). In a user study the LMC was tested in context of an educational environment using pointing gestures and object manipulation gestures combined with the HMD Oculus Rift to see if the LMC is a suitable input device. Additionally to the subjective evaluations of the test users, accuracy measurements were executed to present some numbers. It turned out that most gestures feel natural, but the LMC lacks accuracy in this specific case.

Keywords
user study, pointing gestures, object manipulation gestures, head mounted display, hand tracking device

1. Introduction

Virtual reality environments are sometimes displayed via Head Mounted Displays (HMD). Using an HMD the user does not see the real world. Not seeing the environment can make it difficult to use input devices that require to be touched like mouse and keyboard. Not seeing the keyboard increases the probability of input errors and consequentially it takes more time to make the right input. Hence, while wearing an HMD it could be helpful to interact with the virtual world by using haptic input devices with less buttons or tracking devices which do not need to be seen. Among others, input devices used in virtual reality are game controllers, joysticks, data gloves or optical tracking systems (Sherman et al. 2002; Chow 2008; Lu et al. 2012).

Recently, the Leap Motion Controller (LMC) was developed as an inexpensive 3D hand and finger tracking device. This paper investigates whether the LMC could be properly used in combination with an HMD. Since it is difficult to make statements about the general aptitude of an input device, the tests were conducted in a specific manner. A user study about the usability of pointing and object manipulation gestures with the LMC was made. Therefore, an exemplary educational environment was designed to give a reasonable context for this setup. The short idea is that a user can readout information about objects. Based on this information the user can move and rotate them to execute a given task.

The contribution of this paper is that it was shown that the LMC is not a generally good candidate to use for the implemented gestures combined with an HMD. To point at objects the accuracy of the LMC is adequate. Most gestures feel natural although most test users enjoyed using the LMC, moving and rotating objects turned out to be too inaccurate. Probably some problems could be solved through future work to improve the usability like discussed in the evaluation section.

In the next section related work will be addressed. It will be followed by the description of the test environment including the interaction concept. Also, some words about the implementation, hardware setup and hardware problems can be found. Then, the evaluation will be discussed including a comparison with mouse and keyboard, the test execution and the data spreading. Lastly, a conclusion and some ideas for future work will be presented.

2. Related Work

As described in the following section, other papers with similar or related problems exist. For example, Chow (2008) implemented an interaction concept using the Wii Remote (aka Wiimote) combined with an HMD. His goal was to investigate the feasibility of using the Wii Remote for 3D interaction in immersive HMD virtual reality. Usually, it is required to point the Wii Remote at the display or furthermore at the sensor bar near the display. However, he created a system which makes it possible for the user to turn around while using the Wii Remote, in contrast to the approach of this paper where the user cannot turn around while using the LMC. The Wii Remote was used to point at locations. The Wii Remote extension Nunchuck which contains an analog stick was used to translate the virtual camera. Kuntz et al. (2012) wrote a paper about how to build a low-cost home-made virtual reality hardware setup in which they used an HMD combined with the Wii Remote and the Razer Hydra. The setups were tested in games like VR Escape where one has to take and use objects to escape from a lab. Their input devices are less complex than a keyboard because all buttons can be reached without reposition the hand, but one still has to learn the positions of the buttons and one still has to touch the devices in contrast to the LMC. Using optical tracking systems enables the use of natural gestures that do not have to be learned in advance like the pointing gesture (Roth 2001). In comparison to game controllers like the Wii Remote and the Razer Hydra, this could be an advantage in favor of the LMC.

Lu et al. (2012) used a data glove to recognize hand gestures and an additional system to track head and hand positions. They used natural gestures to interact with objects. For example, they used a pointing gesture to select objects and an open hand gesture to deselect them. Thereby, the gestures are related to the gestures used in this project. Apart from that, there are also downsides to their setup: Firstly, gloves need to be put on prior to any further action with the system. Secondly, their setups consists of several components: The hand and head tracking system and the data glove are two autonomous systems leading to a more complicated setup. Opposed to that, the LMC is a plug-and-play system.

In the paper of Kim et al. (2012) a potential candidate to test in combination with an HMD called Digits is described. It is a small camera-based sensor attached to the wrist that optically images a large part of the user's bare hand. It recognizes hand gestures that can be used to trigger actions. The accuracy is comparable with a data glove. The drawback is that there is no hand position tracking included leading to a more limitations of gestures compared to the LMC.

Pointing in virtual reality is a subject of several researchers. Some of them are using tracking systems with a much bigger tracking area than available in this approach, like Hyung-O et al. (2008), Nickel et al. (2003) or Moeslund et al. (2002). In some of these approaches the entire arm and its pointing direction are tracked. By that, the problem of exiting the tracking area while looking and pointing at an object does not figure prominently. In this approach it is possible to physically look in a direction where you cannot point at without exiting the tracking area. The problem is that more detailed hand gestures cannot be recognized. For example, they do not differentiate between a closed and an open hand.

A more general paper about hand gesture interaction was written by Sturman et al. (1989). They used a data glove to research several hand gestures. Some tests were conducted, such as using the hand as a button, as a valuator (like a slider) or as locator and picking device for pointing and object manipulation in 3D space. They designed an interface to use the hand as an expressive input medium that is independent from the input device.

3. Concept

In the introduction an example about a reasonable context for pointing and object manipulation was given. In this chapter the example is described in more detail.

One could imagine an educational environment where objects can be moved to transpose learned circumstances. For example, the student was taught how to assemble a machine. To check if a student understood his previous lesson, he or she is given the task to build parts of the machine.

Another feature is the possibility to readout information about the objects. For example, if some batteries are placed in the virtual world, there could be added some information about their capacity which the user could readout. That capacity information can be used to choose the right battery for the machine the users is building.

In the test environment that was created for this project the following interactions are possible: move the virtual camera (parallel to the ground), rotate the virtual camera, move objects, rotate objects and show information about objects. To perform these interactions it was tried to use the most natural gestures possible.

The user controls a visible ray that starts from his virtual hand and goes through a point that is positioned by the user's main hand. With the open hand the user is able to hover/select objects by hitting them with the ray (Fig. 1 a). This way of selecting objects was inspired by the ray-casting technique (Bowman et al. 1997).

If the user uses the pointing gesture by extending the thumb and the pointing finger, information about the object will be displayed (Fig. 1 b).

In addition, if the user closes his hand while an object is hit by the ray, the object gets attached to his hand position. By moving his hand the user can move the object like he would do it in the real world (Fig. 1 c).

It is possible to change the interaction mode from moving objects to rotating objects by rotating the open hand a little to the right (Fig. 1 d). In the rotation mode two axes are displayed, to which the selected object can be rotated around. The user can rotate the object by moving his closed main hand up/down and left/right (Fig. 1 e). It is also possible to change back to the moving mode by rotating the open hand to the right again.

The user can move the virtual camera by closing his off-hand and pull the camera forward by moving his closed hand back. Like one would grab the ground and pull one's body forward. That works analogously to the other directions (backwards, left, right).

The tracked head movements by the Oculus Rift are used to rotate the virtual camera. Additionally, it is possible to rotate the camera around the yaw-axis by moving the main hand to the left and right borders of the tracking area. This additional way is needed because if the user is sitting he cannot turn his head 360° around.

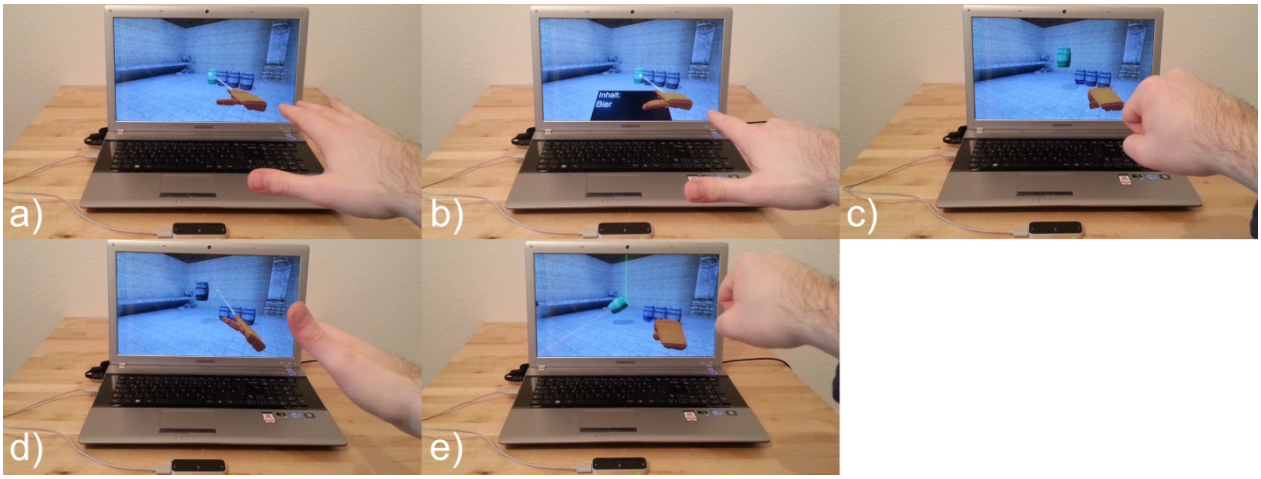

Fig. 1: a) Hovering/selecting an object. b) Pointing at an object to get information. c) Grabbing and moving an object. d) Switching object interation modes between moving and rotating. e) Grabbing and rotating an object.

4. Implementation

The test environment was created with the Unreal Development Kit (UDK) which is a development interface for the Unreal Engine 3 (Epic Games 2014). It was chosen due to its native support for the Oculus Rift, the detailed documentation and the active community. The LMC is not originally supported by the UDK but an open source integration was available (Lamarche 2013). The LMC integration was used to get the palm and finger positions of the user. The positions of palm and fingers were used to recognize the different gestures. For instance, if there were no finger positions recognized, the hand is closed and can trigger corresponding actions like grabbing an object. If five fingers were recognized the hand must be open leading to releasing an object. To check if the thumb is visible a finger had to be in a specific position related to the palm. The thumb had to be recognized for the pointing gesture. Also the movements of the palm were tracked, which were used to control a grabbed object or the pointing ray for example.

Left and right hand were differentiated by the first palm position recognized. If a palm gets recognized on the right side of the LMC recognition area it is by definition the palm of the right hand. This does not have to be correct at all times but it turned out to work nearly always. The decision whether it is the right or the left hand was not made by the side the thumb gets recognized because the user is able to enter a closed hand in which no thumb is visible.

4.1 Hardware Setup

As mentioned in the introduction, the LMC is used to track the hands of the user. The LMC is a small USB device designed to be placed on the table facing upward. Looking at the device, three infrared LEDs and two cameras can be seen inside. Using the recorded pictures, 3D positions of the palms and fingers are getting synthesized. Additionally, the rotation of palms and fingers get calculated. With the used configuration the LMC worked with 115 frames per second. Leap Motion (2014)

In this paper an HMD was used to display the virtual world. The used HMD is the first developer version of the Oculus Rift. The following data is taken from the Oculus SDK Overview (Oculus VR 2014): Inside the Oculus Rift is a 7 inch screen with a resolution of 1280x800. Each side of the screen shows the picture of the virtual world for the corresponding eye. This results in a resolution of 640x800 for each eye. The user gets a field of view of about 110°. Also, the head orientation is tracked by the Oculus Rift that can be transferred to the virtual camera. Using that, the illusion of moving the head inside the virtual world can be created.

4.2 Hardware Problems

The LMC and the Oculus Rift do not work perfectly. The following four recognizing problems with the LMC were observed. Sometimes finger positions disappear from one frame to another. Especially the tracking of the closed hand seems to be inaccurate which leads to jittering that had to be compensated. The tracking of the open hand is much more accurate. As mentioned above, the palm rotation gets calculated too, but the rotation changes (up to 30°) depending on how many fingers are visible. Especially if the hand is closed, the rotations recognized by the LMC are inaccurate. Originally it was planned to implement the pointing gesture with only the pointing finger visible. The thumb had to be added to the pointing gesture due to the issue that the LMC sometimes recognizes a finger at the position the pointing finger would be, even if the hand is closed. So it could not be differentiated between a closed hand and a hand with only the pointing finger extended. In this implementation a closed hand is recognized, if one or zero fingers are visible.

The head tracking of the Oculus Rift is accurate and fast but still leads to motion sickness for some people. The most prominent problem with the Rift is the low resolution display and the blurring generated by the lenses the user looks through. Therefore, reading text is difficult, if it is not centered in the Oculus Rift screen. To make the object information readable, the size of the font was chosen fairly large.

5. Evaluation

5.1 Comparison with Mouse and Keyboard

To get a better sense of the accuracy and the feeling of the LMC it was compared to mouse and keyboard. The control of the interactions by mouse and keyboard are described in the following.

The user controls the view perspective of the virtual camera via mouse movements (like in modern first person shooters).

Objects in the middle of the screen get hovered to indicate interaction-possibility. By performing a left mouse click information about the hovered object is getting displayed.

Is the user holding the right mouse button down while aiming at an object that object gets attached to the center of the screen. By changing the camera perspective the position of the object changes too. Additionally, the user is able to change the distance between him and the object by holding the buttons Q or E.

If the user holds the left and the right mouse button down, he is able to rotate the object by moving the mouse. While rotating an object the camera perspective is locked.

The user can move the virtual camera by holding the buttons W (forward), A (sideward left), S (sideward right) and D (backward).

5.2 Test Execution

The task to perform by the test users is to sort barrels into shelves. One shelf is reserved for food and the other for drinks. The users first have to readout information about the barrels in order to know what is inside them. After the content is known the users can place the barrels into the specific shelves. However, before putting them down, the

barrels must be rotated to be in a horizontal position. The user test was designed in a way that ensured every test user performed each gesture.

Every user executed the described task two times. One time with the LMC and Oculus Rift as input devices whereby the Oculus Rift is also the visual output device. And the other time with mouse and keyboard as an input device and a normal display as a visual output device. The setups, the users started with were alternated. In both setups the users were sitting.

The task was explained to the users by a test leader. He sorted the first barrel into the right shelf. While doing this he explained the controls to the users. Afterwards the users could start the task and sort in the three remaining barrels.

The 23 users, 2 female and 21 male, including 6 left handed and 17 right handed were mainly professors or students. The age spreading was: 4 times under 21 years, 5 times between 21 and 25, 3 times between 26 and 30, 2 times between 31-40 and 9 times over 40.

After they finished the task with both setups they completed a survey that included 23 questions. 12 questions were based on direct comparison of the two different interaction concepts whereby the users could evaluate the concepts via the semantic differential scale. Also, some questions about the comfort of LMC and Oculus Rift were asked and could be evaluated in 7 steps. Additionally, the users were able to write down comments about the setups.

5.3 Data Spreading

Following conclusions were made evaluating the data collected in the earlier described user test. The spreading of the data and the associated questions can be found in Fig.2.

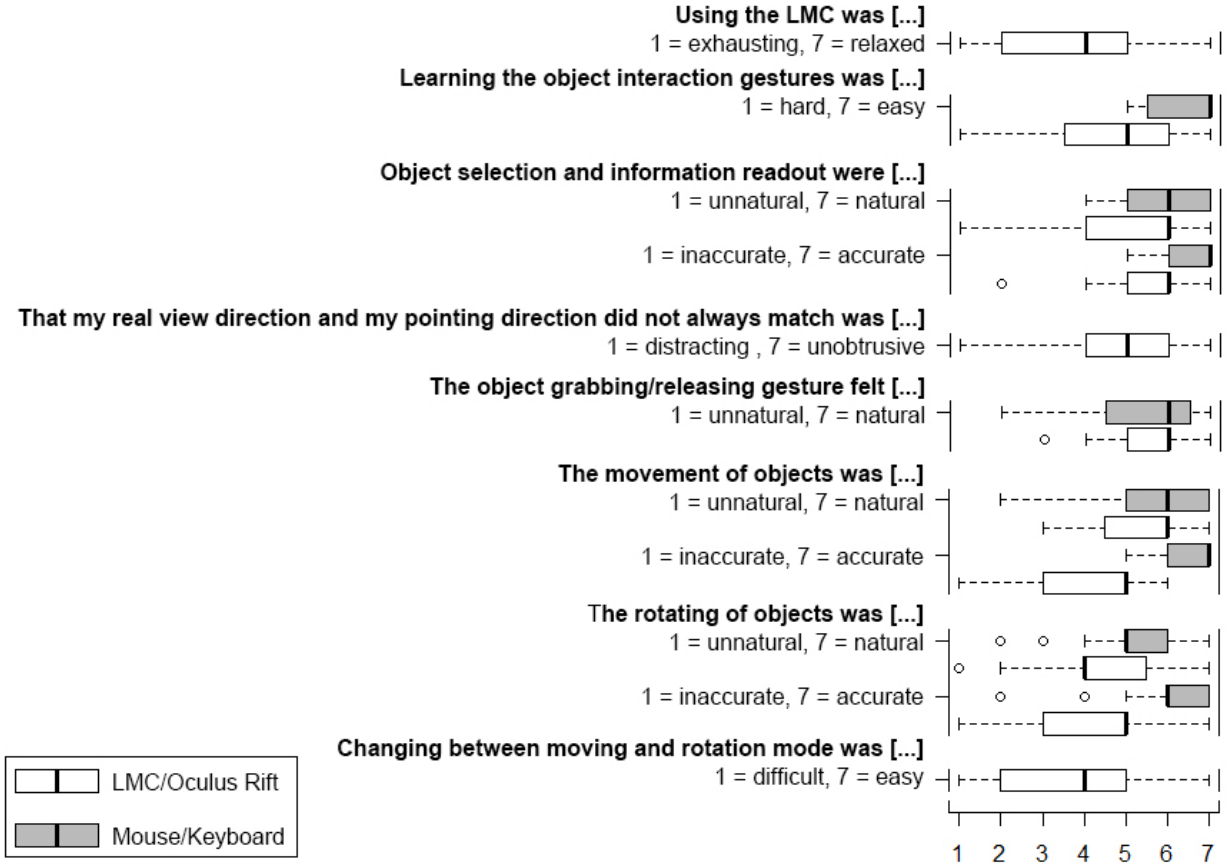

Fig. 2: Spreading of the evaluations shown via boxplots.

5.4 Survey Discussion

In the survey the two setups were compared directly what leads to two paired data populations. Therefore, to evaluate the significance of the analogy of the two populations the Wilcoxon–Mann–Whitney test was used (marked

with "WT"). Not only comparisons were made. Some features of the test environment or the hardware should be evaluated absolutely and not compared to something else. In a range between 1 and 7 the values 5-7 are interpreted as positive and the rest as negative. To test the significance of the deviations of the two categories, the binomial test was used (marked with "BT").

The users were asked to evaluate how it felt to use the LMC, whether it was exhausting or relaxed. The results tend to "exhausting" or something between "exhausting" and "relaxed". A general statement cannot be given (values 5-7 = relaxed: BT p-value = 0.895; values 1-4 = exhausting: BT p-value = 0.2024). Nevertheless, one has to consider that the usage-duration of the LMC was about 10 minutes. It is conceivable that people get more exhausted after a longer time using it.

The gestures are easier to learn with mouse/keyboard than with LMC/Oculus Rift (WT p-value = 0.0003811). But one has to keep in mind that the mouse is way more common than the LMC and most of the mouse gestures can be found in many other applications.

The accuracy of the object selection/pointing at objects was received well with both setups (values 5-7 = accurate: BT p-value < 10^{-4} for mouse, BT p-value = 0.0013 for LMC). But the selection with the mouse is more accurate than with the LMC (WT p-value = 0.000443). Additionally, the users think the object selection feels more natural with the mouse (WT p-value = 0.01375). Some users mentioned that the pointing gesture with the pointing finger and the thumb feels unnatural because they normally point without extending their thumb. The decision to add the thumb was made due to recognition problems with the LMC. In order to fix this, a way must be found to compensate the recognition problem.

The users were also asked if it is distracting or unobtrusive that pointing and viewing direction do not match at all times. The evaluation showed that most people where in-between, therefore none of the statements could be confirmed by the statistical test (values 5-7 = unobtrusive: BT p-value = 0.105; values 1-4 = distracting: BT p-value = 0.9534).

The way an object is grabbed and released feels natural on both setups (values 5-7 = natural: BT p-value = 0.01734 for mouse, BT p-value < 10^{-4} for LMC) but based on the test results no statement could be given that one is better than the other (LMC more natural: WT p-value = 0.2747; mouse more natural: WT p-value = 0.7513).

Also the movement of objects felt natural on both setups (values 5-7 = natural: BT p-value = 0.0013 for mouse, BT p-value = 0.01734 for LMC) and again none of them could be confirmed better than the other (LMC more natural: WT p-value = 0.6509; mouse more natural: WT p-value = 0.3643) but the accuracy of the object positioning was better with mouse and keyboard than with the LMC (WT p-value < 10^{-4}). There are some things that could explain the inaccurate object movement with the LMC. Firstly, the arm of the user still moves a little, even if he tries to hold it at the same position. Secondly, the LMC creates jittering especially, if the hand is closed. Thirdly, and often criticized in the comments: While the user is opening his hand the LMC recognizes a strong palm movement even if the palm does not move. So every time the user opens his hand the object moves a little and is not positioned where the user intended it to be.

The rotation of objects felt natural with mouse (values 5-7 = natural: BT p-value = 0.005311) but not with the LMC (values 5-7 = natural: BT p-value = 0.6612). Consequently, the rotation with the mouse feels more natural than with the LMC (WT p-value = 0.0388) and is additionally more accurate (WT p-value = 0.0001611). The inaccuracy with the LMC is probably caused for the same reasons the object movement is inaccurate. The hand movements for rotating an object and moving an object are similar.

A statement about the usability of the gesture to change between moving and rotating mode cannot be given, since the probability of error is too high (values 5-7 = easy to use: BT p-value = 0.9534; values 1-4 = hard to use: BT p-value = 0.105).

In sum, users enjoyed the LMC and Oculus Rift more (57%) than with mouse and keyboard (30%). Some people did not enjoy any of them (13%). But that values could not be used to generally say that LMC and Oculus Rift is more fun than mouse and keyboard.

5.5 Measurement Discussion

Not only the subjective evaluation of the accuracy was made. The accuracy was also measured in the application itself. Therefore, one user was asked to place a barrel at a specific location. The distance to the target location was measured before and after the user released the barrel. Subsequently, the user was told to point at the center of a

square. The distance between pointing position and square center point was measured as well. These two tasks were executed 15 times with LMC and 15 times with mouse and keyboard.

Since there were two data populations to compare, the Wilcoxon–Mann–Whitney test was used again (marked with "WT"). To get a sense of the units used in the UDK some dimensions are described in the following: The barrel had a caliber of 48 units and was 64 units tall. A barrel is about half as tall as the users avatar.

The average distance between barrel location and target location was 2.09 units with LMC before the barrel was released and 22.05 units right after. The average distance with mouse and keyboard was 1.78 units, whereby there was no difference between before and after the release. Hence, it was shown that object movement is more accurate with mouse and keyboard than with LMC after releasing the object (WT p-value < 10^{-4}). Before releasing the barrel it could not be proven that mouse and keyboard are more accurate than the LMC (WT p-value = 0.1947).

The average distance between pointing location and target location was 2.37 units with LMC and 1.82 units with mouse and keyboard. In general, the statistical test could not show that pointing with a mouse was more accurate than with the LMC (WT p-value = 0.1147).

In the test environment inaccuracy can be subjectively recognized from a shift of about 4 units. By using the binomial test (market with "BT") sorting values into the two categories; accurate (values < 4) and inaccurate (values >= 4) the following conclusions can be made: With the LMC object movement is accurate before releasing the object (BT p-value < 10^{-4}), it is inaccurate after releasing the object (BT p-value < 10^{-4}) and the pointing gesture is accurate (BT p-value: 0.01758). With mouse and keyboard both object movement (BT p-value = 0.003693) and the pointing gesture (BT p-value < 10^{-4}) are accurate.

This shows that using the LMC the release gesture is the main source of inaccuracy while placing objects.

6. Conclusion and Future Work

Compared to mouse and keyboard the LMC is not the better choice combined with an HMD with the used gestures, but not generally bad either. Most gestures felt natural and most people enjoyed using the LMC combined with the Oculus Rift. Especially the pointing gesture should be highlighted since it was the only gesture on the LMC that felt natural and was accurate. Other gestures like the grabbing/releasing object gesture and object movement gesture felt natural but lacked in accuracy. The object rotating gesture was not natural nor accurate. As a result, the biggest disadvantage of the LMC in this setup is the inaccuracy while moving and rotating objects what is caused by the displacement while releasing the objects.

A general inaccuracy while moving objects and pointing could not be shown by evaluating the data collected by measurement. Rather, they were evaluated accurate except for the release gesture with the LMC leading to displacements of objects.

Future work should address the problem of displacing objects while releasing them. To circumvent that problem a docking system could be implemented so that the users do not have to position the object accurate, they just have to move it near a docking point so the object snaps into the right position. Another option would be to implement a different gesture to release the object. For example, a gesture with the other hand so the object is not influenced by the releasing gesture. But that could lead to an unnatural feel of use.

Because some of the test users criticized the pointing gesture, a new way should be found to enable the pointing gesture without the need of extending the thumb.

Since the object rotation gesture and mode changing gesture could not convince either, a further step would be to replace them.

7. References

Bowman A, Hodges F (1997) An Evaluation of Techniques for Grabbing and Manipulating Remote Objects in Immersive Virtual Environments. Proceedings of the 1997 Symposium on Interactive 3DGraphics. 35-ff.

Chow YW (2008) The Wii Remote as an input device for 3D interaction in immersive head-mounted display virtual reality. Proceedings of IADIS International Conference Gaming. 85-92

Epic Games (2014) Free Game Engine for Indie Game Development | UDK Unreal Developer's Kit. http://www.unrealengine.com/udk/. Accessed 12 February 2014

Hyung-O K, Soohwan K, Sung-Kee P (2008) Pointing gesture-based unknown object extraction for learning objects with robot. Control, Automation and Systems. 2156-2161

Kim D, Hilliges O, Izadi S, Butler AD, Chen J, Oikonomidis I, Olivier P (2012) Digits: freehand 3D interactions anywhere using a wrist-worn gloveless sensor. Proceedings of the 25th annual ACM symposium on User interface software and technology (UIST '12). 167-176

Kuntz S, Cíger J (2012) Low-cost and home-made immersive systems. The International Journal of Virtual Reality. 1–9

Lamarche H (2013) LeapUDK. https://bitbucket.org/HugoLamarche/leapudk. Accessed 12 February 2014

Leap Motion (2014) Buying a Leap Motion Controller : Leap Motion Support. https://leapmotion.zendesk.com/entries/39268303-Buying-a-Leap-Motion-Controller. Accessed 13 February 2014

Lu G, Shark LK, Hall G, Zeshan U (2012) Immersive manipulation of virtual objects through glove-based hand gesture interaction. Virtual Reality. 16:243-252

Moeslund TB, Störring M, Granum E (2002) A natural interface to a virtual environment through computer vision-estimated pointing gestures. Gesture and Sign Language in Human-Computer Interaction. 2298:59-63

Nickel K, Stiefelhagen R (2003) Pointing Gesture Recognition Based on 3D-tracking of Face, Hands and Head Orientation. Proceedings of the 5th International Conference on Multimodal Interfaces. 140-146

Oculus VR (2014) Oculus SDK Overview. http://static.oculusvr.com/sdk-downloads/documents/Oculus_SDK_Overview.pdf. Accessed 10 February 2014

Roth WM (2001) Gestures: Their role in teaching and learning. Review of Educational Research. 71:365-392.

Sherman WR, Craig AB (2002) Understanding virtual reality: Interface, application, and design. Elsevier. 452-453

Sturman DJ, Zeltzer D, Pieper S (1989) Hands-on interaction with virtual environments. Proceedings of the 2nd annual ACM SIGGRAPH symposium on User interface software and technology (UIST '89). 19-24