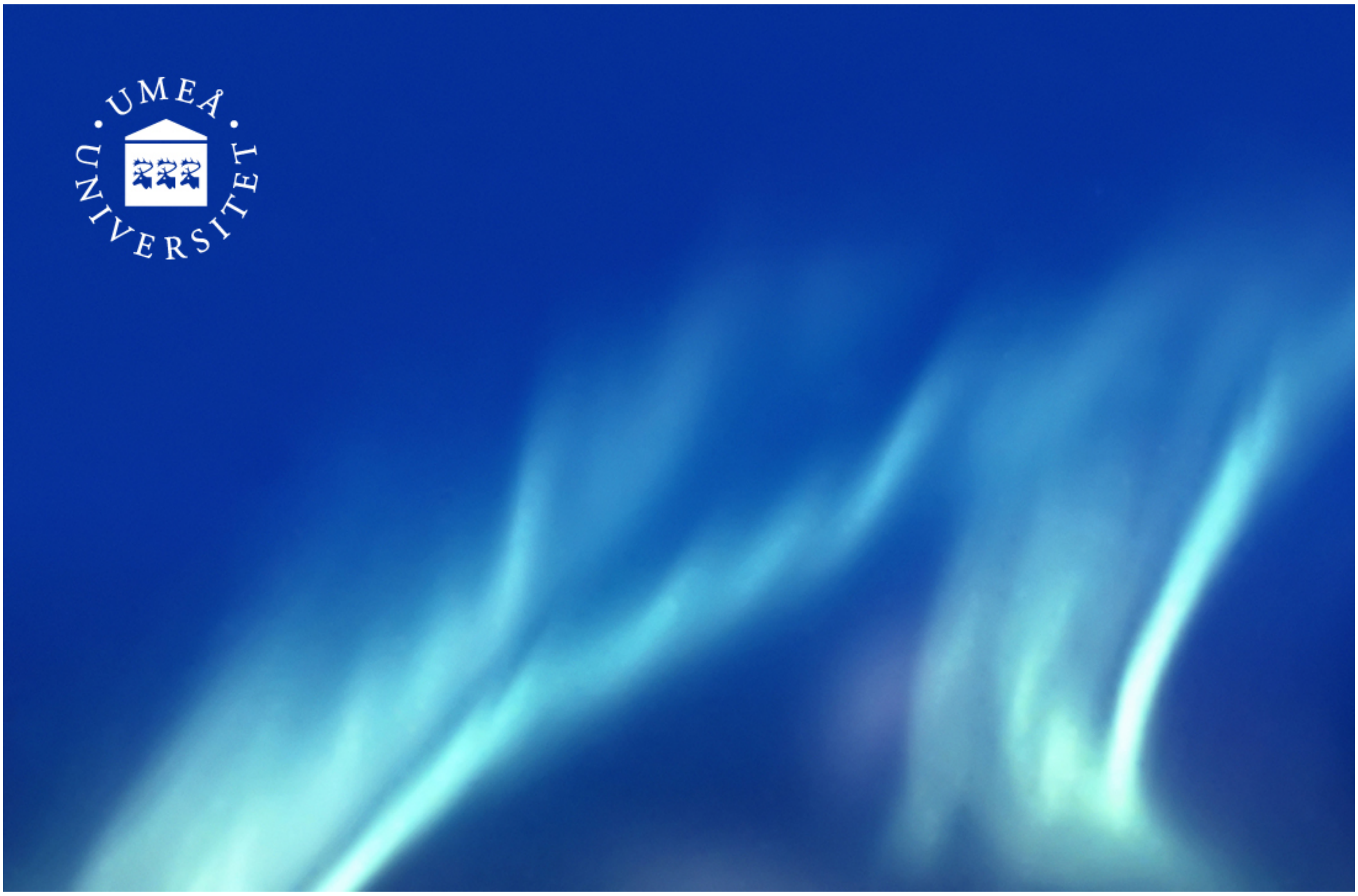

Master Thesis Report

Place the title (maximum 2 rows)

Place the subtitle (maximum 2 rows)

David Bergvik

February 10, 2017

Student
Fall 2017
Master Thesis, 30 ECTS
Master of Science in (state your degree), 300 ECTS

1 Introduction

This is a section of this chapter. use subimport subsections/filename to add subsections

2 Thoeretical Framework

This is a section of this chapter.

2.1 Interaction Tools in VR

There is a lot of parameters in the virtual world to take into account when deciding on a selection tool and technique. Some of the tools and techniques that exists for VE will be explained.

2.1.1 Raycast techniques

Multiple studies have concluded that in an environment with a sparse selection of objects with a volume that is not too small, using a raycast is most really fast and reliable. [insert raycast (pointer) references here] When some of these parameters change however, the raycast tool with a "laser-pointer" technique experiences more issues. In an environment that contains alot of objects in a small space, the error rate rises. This factor is multiplied when movement is added to the object (typical for games).

Accordning to [Dense and Dynamic 3D Selection for Game-Based Virtual Environments] there are better ways to perform object selection in a more complex and dynamic environment by tweaking this concept. By using techniques that are designed for dynamic and cluttered environment the speed and error rate can be reduced. Two of these techniques are 'zoom' and 'expand'. On first selection the surrounding area of the selected object is enhanced to simplify the selection.

A big problem with pointing is trembling of the hand and twitches that occour when user tries to select an option. This has been given the name "Heisenberg effect" and is the cause of new interaction issues:

- user dissatisfaction due to increased error rates,
- discomfort due to the duration of corrective movements, which in the absence of physical support require an additional physical effort, and
- unconfidence on which object will be selected after triggering the confirmation

[Improving 3D Selection in VEs through Expanding Targets and Forced Disocclusion]

2.2 Problems with interactions in VR

[INTRO MISSING]

2.2.1 Interaction Technique issues

[INTRO MISSING] [New Directions in 3D User Interfaces] presents four ways that the majority of interaction techniques exhibit generality:

- Application- and domain-generality: The technique was not designed with any particular application or application domain in mind, but rather was designed to work with any application.
- Task-generality: The technique was designed to work in any task situation, rather than being designed to target a specific type of task. For example, the design of the ray-casting technique does not take into account the size of the objects to be selected and becomes very difficult to use with very small objects (Poupyrev et al. 1997).

- Device-generality: The technique was designed without consideration for the particular input and display devices that would be used with the technique. Often techniques are implemented and evaluated using particular devices, but the characteristics of those devices are not considered as part of the design process. For example, the HOMER technique (Bowman and Hodges 1997) is assumed to work with any six-degree-of-freedom input device and any display device, but all of the evaluations of this technique have used a wand-like input device and a head-mounted display (HMD).
- User-generality: The technique was not designed with any particular group of users or user characteristics in mind, but rather was designed to work for a "typical" user.

2.2.2 Occlusion problem

A problem with interactions in VR that has a very small significance on screen-based UI is occlusion. Since the user interacts and moves in a VE in 3D and with 3D objects, the possiblity of objects blocking each other. To solve this the user can move arounid in the virtual space and hopefully getting an angle that occludes the object, or use a selection tool that can be bent around the first object or pass through it. [Large Scale Cut Plane:] offers a different solution, where the user can "slice the environment and hide it in order to get access to the desired object. This method were preferable from the standard method which is to move and find a better angle.

2.2.3 Human body limitations

[INTRO MISSING] Fatigue is one of the biggest problems with VR [A survey of 3D object selection techniques for virtual environments]. Selection techniques are more severe on arm and wrist strain/pain while navigation can induce simulation sickness.
Physical reach is also a big problem when interacting in a virtual environment. It limits the interactionspace to the length of the user's body (most often arms).

2.2.4 Physical Space

The journey to a virtual environment using a portable headset does not include a vast infinite empty physical space to move around in. This causes problems when users are imerged as they cannot see the physical objects in the real world which can cause injuries.

2.3 Interface Design

Interfaces in a virtual environment comes with a new set of challenges when compared to a traditional interface designed for a screen. If these challenges are neglected the virtual experience for the user might be troublesome.
When desigining an interface it's important to evaluate the speed of selection. Fitts' law is a fundemental and proven way to evaluate pointing to real-world objects by measuring the distant to the object and its size. [Fitts, P.M., 1954. The information capacity of the human motor system in controlling the amplitude of movement. Journal of Experimental Psychology 47, 381–391.]
This is however based on a real-world scenario which does not translate into a virtual environment were the user need a tool to interact with objects. Despite these differences, Fitts´ law can be applied to pointing in a virtual environment using following formula [Evaluation of mouse, rate-controlled isometric joystick, step keys, and text keys for text selection on a CRT]

$$ID = log(2) * (2 * D/W) \tag{1}$$

where ID is the index of difficulty, D is the distance from startingpoint to the middle of the correct target.W is the width of the target (caluculated on the axis where the pointer will travel).

3 Methodology

This is a section of this chapter. use subimport subsections/filename to add subsections

4 Results

This is a section of this chapter. use subimport subsections/filename to add subsections

5 Discussion

This is a section of this chapter. use subimport subsections/filename to add subsections