Prototyping a gesture controlled Head Up Display using a Leap Motion

Daniel Brand
University of Salzburg
Salzburg, 5020, AUT
Daniel.Brand@stud.sbg.ac.at

Kevin Büchele
University of Salzburg
Salzburg, 5020, AUT
Kevin.Buechele@stud.sbg.ac.at

Abstract
In todays cars, the design of a Head Up Display (HUD) and the arrangement of its components are rather fixed. This paper provides an approach how a Leap Motion can be used to achieve a gesture controlled HUD. Furthermore, some aspects that need to be considered when implementing a modifyable HUD for cars are discussed.

Author Keywords
LeapMotion; HUD; Head Up Display; Automotive; Gesture recognition

Introduction
During the last years, the HUD has been established in middle and high class cars. However, Head Up Displays are rather static, in particular, the position of each element in the display is fixed.

Nowadays, users of any devices appreciate an interface which can be modifyed. Since buttons or similar controls may distract the driver from looking at the road, these types of controlling the interface are rather bad choices.

After doing some research, we came to the conclusion that a touchless control (similar to other approaches to control a infotainment system) may be a reasonable goal we want to achieve.

A classic device to implement touchless controls is the Leap Motion. We use the Leap Motion and furthermore gestures which are easy for the user and also easy to be tracked by the device. In this paper, we will discuss problems and issues we have faced regarding the position of the Leap Motion, which gestures to use and more important which settings/requirements have to be fixed in order to get the system working smoothly. By the end, we can select items within a HUD without any other devices such as a touchpad, etc.

The system's environment

The Leap Motion is a classic sensor device to track the hands of a user and therefore his gestures. The detection of the hand works with an infrared camera which has the advantage that environment light which would generate noise in the information are reduced. The sensor recognizes each finger including bones and joints, which gives us the possibility. So far, our system has been configured to work with a right hand only. For now, there is no support for left handers.

Our HUD will be simulated with the help of the JavaFX framework. In order to test which settings fit the most for this use, we are able to dynamically display 3-5 elements in our HUD, simply represented with numbers between 1 and 3 (respectively 5). Certain gestures above the Leap Motion will cause an interaction with the JavaFX window. The Leap Motion is set up to track 2 major gestures: pointing at a certain field (see figure) and selecting it by moving the pointing finger slightly in direction to the HUD. The current marked element will get a blue background, while the selected element will be shown with a green background. In the case of our prototype, the Leap Motion Listener and the display are running on the same computer.

The angles between finger, Leap Motion and HUD have to be ajusted accordingly. This is because we are using the interface with the right hand, more specificly the right pointing finger, only. Obviously, the anatomy of our hand allows a slightly bigger movement range of the finger to the left instead of the right. This needed some testing, and we came to the conclusion that the angles have to be ajusted within the program in order to achieve a satisfying control of the HUD.

Test setup

Our test setup was built up in the driving simulator of the HCI-Center in Salzburg. A small netbook running eclipse executes the program. The Leap Motion which is plugged into this netbook is placed right behind the steering wheel. It has to be mentioned that we did not use a standard sized steering wheel. Instead, we used a more flattened steering wheel which is quite similar of modern (self driving) cars such as Tesla. Using a standard steering wheel might not lead to similar pleasant results.

Each test person needs to complete two test sessions. The difference between this sessions are the amount of elements in our HUD. After some pre-tests, we now only include test sessions with three and four elements. Using five elements did not lead to pleasant results since selecting one out of five elements was way too difficult.

Each session consists of 15 steps. In each step, the program shows a random number in a small window below the HUD. The user now has to try and select the shown element by moving the pointing finger accordingly (left or right). If the desired element is marked, the user can then select it by moving the pointing finger slightly forward. After successfully (or not) selecting an element, the test sequence is continued with the next element.

Between each step, the user has to put his hand back to the steering wheel again. This makes sure that his hand does not keep on pointing onto the selected element which would make the next selection easier if the same element has to be selected. Furthermore, we think it is more realistic when the user's hand has to be moved from the steering wheel to the HUD.

Throughout the whole test, the user also has to keep driving with the driving simulator. We want our test setup to be as realistic as possible.

The result of each step is recorded and (after completing the sequence) summarized in a log file. The logfile provides detailed test results in order to validate our prototype. The first few lines give a summary about basic settings and results such as implemented angles, test sequence, touched elements in the test sequence and the duration of the whole test. The second bigger a part of the log file gives an exact result about which angle was hit after each step (with respect of the angle window of the element). Also an error rate over all selected (or missed) elements is written into the logfile.

Test results

tbd