Haptic Feedback Through Weight Shifting

Hendrik Schneider, Robert Scheibe, Dipl. Inf. (FH) Jan Hochstrate, Prof. Dr. Bernd Fröhlich
Bauhaus University Weimar
Virtual Reality Systems group
Weimar, Germany
{hendrik.schneider,robert.scheibe,jan.hochstrate, bernd.froehlich}@medien.uni-weimar.de

Abstract

This paper introduces an alternative possibility to create haptic feedback through weight shifting. Our system is based on a simple electro mechanic construction to shift a weight along two parallel guide bars.
We integrated this setup into a sample scenario including a handheld input device and an application scenario. The idea is to develop a fork lift simulation. The user controls a fork lift truck through a virtual environment using the haptic feedback input device.

1. Motivation

Most force feedback systems use motors to generate direct haptic feedback. These systems need a stationary reference, which results in a limited workspace. In virtual reality scenarios users are often moving around to explore the virtual environment from different perpective. A solution are handheld devices, which provide some type of haptic feedback. One example for tactile feedback are small vibrating motors, used in CyberGlove devices or build into a tracked handle[Schoenfelder et al.].

2. Overview

We developed an alternative solution to create a haptic sensation by utilizing the gravitation. Figure 1 shows the main idea which is based on shifting the centroid of an input device to create haptic feedback. The U-shaped construction contains two parallel guide bars. The steel weight can be moved along these guide bars.
The rest of the papers is organized in the following manner. Section 3 provides further detail to hardware and software issues. Part 4 deals with the sample application scenario, and section 5 concludes the paper.
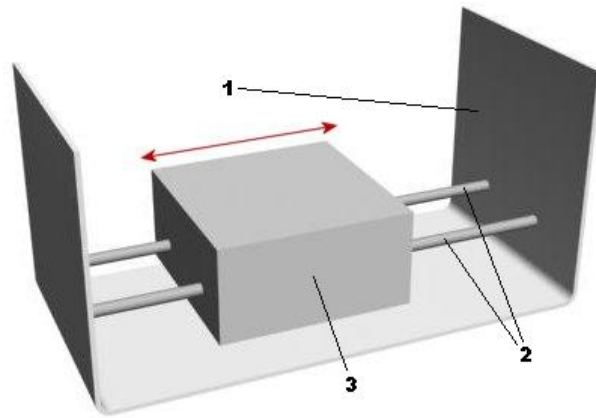
Figure 1. Weight shifting unit: 1 - U-shaped chassis, 2 - guide bars, 3 - steel weight

3. Technical Details

3.1. Hardware Decicions

We used the ATMega32 microcontroller for developing our prototype. We needed a analog-digital converter and UART interface to communicate with the VR application. So the features of the ATMega32 we were interested in are:

- Programmable Serial USART

- 8-Channel 10bit ADC

- 32 IO Pins

The microcontroller is used to measure all incoming data, transfer the processed information to the VRPN server and execute commands send from our application.
Figure 4 shows the mainboard of the device, housing the microcontroller, motor drivers and module-connection pins.
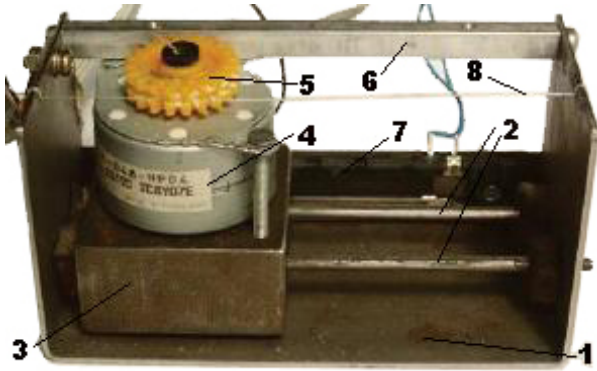
Figure 2. Weight shifting unit: 1 - U-shaped chassis, 2 - guide bars, 3 - steel weight, 4 - stepping motor, 5 - modified gear wheel, 6 - stabilization bar, 7 - modified potentiometer, 8 - nylon string

Figure 3. Cross section of the input device: 1 - weight shifting unit, 2 - analog joystick, 3 - motor-driven studio fader, 4 - status indicator LEDs, 5 - RS232 serial interface

To ensure flexibility every part in the prototype is built as a single module, which is then attached to the mainboard.
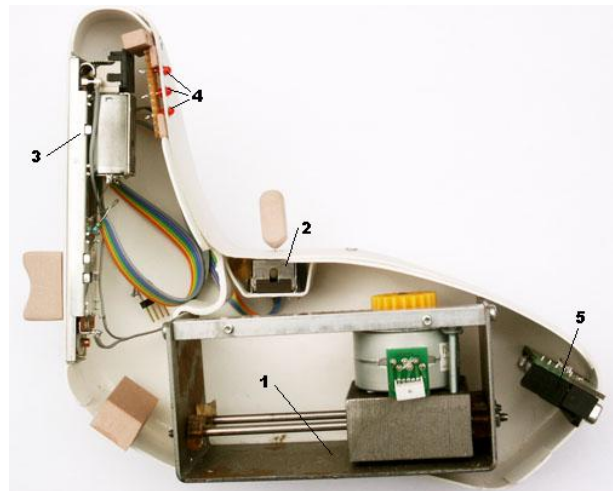
We decided to use a stepping motor to solve the problem of movement. This motor is directly mounted on the massive steel weight. The whole unit is moveable along the steel guide bars, which are mounted in between the U-shaped chassis. For shifting the weight unit a fixed reference point is needed. We use a nylon string which is connected to the static parts of the construction and wrapped around the motor wheel.

For measuringg the position of the steel weight we used a modified potentiometer. The sliding contact of the potentiometer is directly glued to the steel weight.
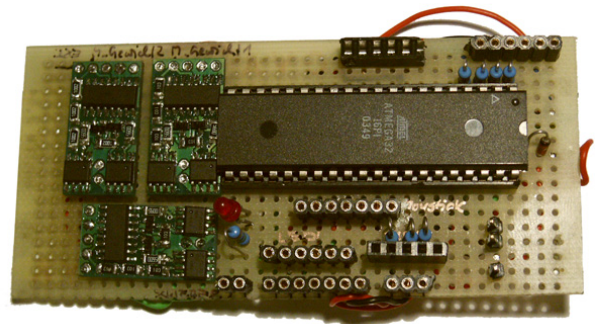
Aside from the weight shifting unit our input device also contains further components as shown in figure 3. The analog joystick is used to steer the fork lift truck in a virtual three-dimensional environment. We chose a motor-driven studio fader to adjust the fork's height. The serial RS232 interface is used for the communication with the vrpn server. Three buttons on top of the device are configured with different functions, such as preset fork positions or enabling the fork lift process. There are also three LEDs, which serve as status indicators.

3.2. Software-Framework

We used the Avango^{TM}VR-System [1] with the Vortex [2] library for simulating physical behaviour of all objects in the virtual environment. For communication purposes we use the Virtual Reality Peripheral Network [3] framework.


Figure 4. main board with ATMega 32 Micro-controller

Avango is a object-oriented framework for the development of distributed interactive VE applications developed by the Fraunhofer IMK. Its indended to be a flexible framework for fast application delevelopement thru its scheme scripting interface.

The Virtual Reality Peripheril Network (VRPN) [3] framework is a set of different classes within a library and a set of server programs, which forms a network transparent interface between applications and physical tracking systems in a VR application. We used the VRPN framework to ensure communication with the avango application. The communication process is managed by channels. Each channel is either declared to be write, read or write/read accessible.

4. Application Scenario

4.1. General Description

We developed the sample application shown in figure 5 to test our haptic input device metapher.
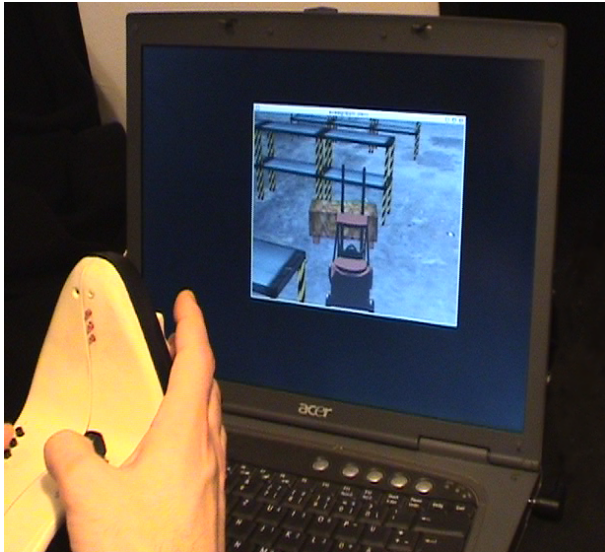
Figure 5. Avango sample application

The user is able to move the fork lift truck in the virtual world by using the analog joystick. He/She can adjust the height of the fork with the studio fader, which is located at the front side of the device. This is necessary for picking up the boxes. If one box is on the fork the weight will be shifted to the front part of the input device. Accordingly the user recognizes a change of the balance point of the device which feels like raising the weight of the input device itself. After releasing the box in one of the shelves the weight in the input device will be shifted to the back part of the device again. In this case the user will have the impression that the input device becomes lighter.
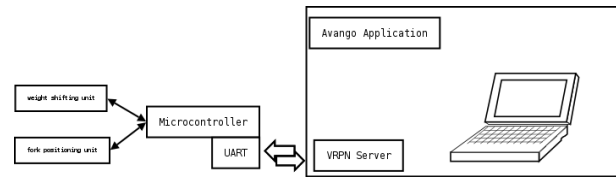
4.2. Implementation in Avango

The application was developed in Avango, using Vortex to simulate realtime physics.

SOME WORDS TO VRPN AND AVANGO
The overall communication diagram is shown in figure 6. We defined three write/read VRPN channels for our device:

- Channel 0: sets the LED on top of the device

- Channel 1: sets the fork position

Figure 6. communication scheme

- Channel 2: sets the weight position

And three ReadOnly Channels:

- Channel 3: joystick x direction

- Channel 4: joystick y direction

- Channel 5: button's state

The channels are accessible from the scheme application script in the following way:

```
(-> atmel 'get-channel 3)
(-> atmel 'set-channel 2 233)
```

5. Conclusions and Future Work

We have developed an alternative solution for providing haptic feedback in a handheld input device. Our approach is based in the idea of shifting the centroid of the input device to create the sensation of picking up an object or dropping it off.
The sample input device shows that our concept works even though there occurred some problems. The feedback from weight shifting process is sensible, but still too weak. Using this technology one has to find a trade-off between size and mass of the weight and the quality of the haptic feedback. Maybe an impulse feedback could be a solution for this problem. The idea is to shift a less heavy weight using a higher velocity.
Many application scenarios for this type of haptic feedback are imaginable. A promising idea is to integrate this technology into a pointing device like it is often used in virtual reality applications. The user can pick up different kinds of objects in a virtual environment. The weight shifting unit can be used to simulate the weight of these objects.

6. Acknowledgements

References

[1] *Avango^{TM}*. `http://www.avango.org`.

[2] Vortex. *Vortex - REALTIME MOTION SIMULATION*. `http://www.cm-labs.com/products/vortex/`.

[3] VRPN. *Virtual Reality Peripheral Network*. `http://www.cs.unc.edu/Research/vrpn/`.