COMPLETELY CONTROLLER FREE INTERACTION IN THE CAVE

Forschungsprojekt Teil A: Analyse

Student: Ilyes Soussi

Supervisor: Prof. Thomas Jung

Hochschule für Technik und Wirtschaft Berlin
University of Applied Sciences

Table des matières

Table of figures:

Introduction

The ongoing advancement and reduction of hardware in the field of information technology has given us a lot of virtual and extended realities in recent years. Nowadays photorealistic computer games are within easy reach and Augmented Reality applications on smartphones are used by many users.

the HTW-Berlin provides such a space for study and research purposes in which a user can immerse himself on three side walls and the ground into a virtual world using 3D stereo-projection.

One of the challenges in the CAVE is to make it completely free of the controller interaction, that means, movements and actions in the cave can be done by only detecting the user's gestures.

In the past, classic input devices such as joysticks or the Nintendo WiiMote have been used for this purpose. Even if these methods of input can be described as intuitively in today's world, they are unnatural and the feeling of immersion isn't fully satisfying, with the help of the two present Microsoft's Kinect, it is possible to capture gestures such as movements and user's actions.

In this project, we will attempt to make these changes and adapt them to the HTW-CAVE.

I. Background

1. Immersive virtual environment:

Immersion into virtual reality is a perception of being physically present in a non-physical world. The perception is created by surrounding the user of the VR system in images, sound or other stimuli that provide an engrossing total environment [Wikipedia, 2016a].

The immersion in computer games is done by the interaction of the player with the virtual world. There are always new types of interaction devices such as Nintendo Wii-mote as well as Microsoft Kinect, all of which have the goal that the player can act as naturally as possible within his virtual environment.

2. Cave Automatic Virtual Environment:

CAVE [1] is a three-dimensional virtual space in which the viewer can move freely. In this room, the walls, floors and ceilings are made of projection surfaces.

The entire virtual movement is monitored by sensors that are located in the room, taking into account the position of the viewer, its viewing angle and perspective. The viewer can view individual objects or images from all sides.

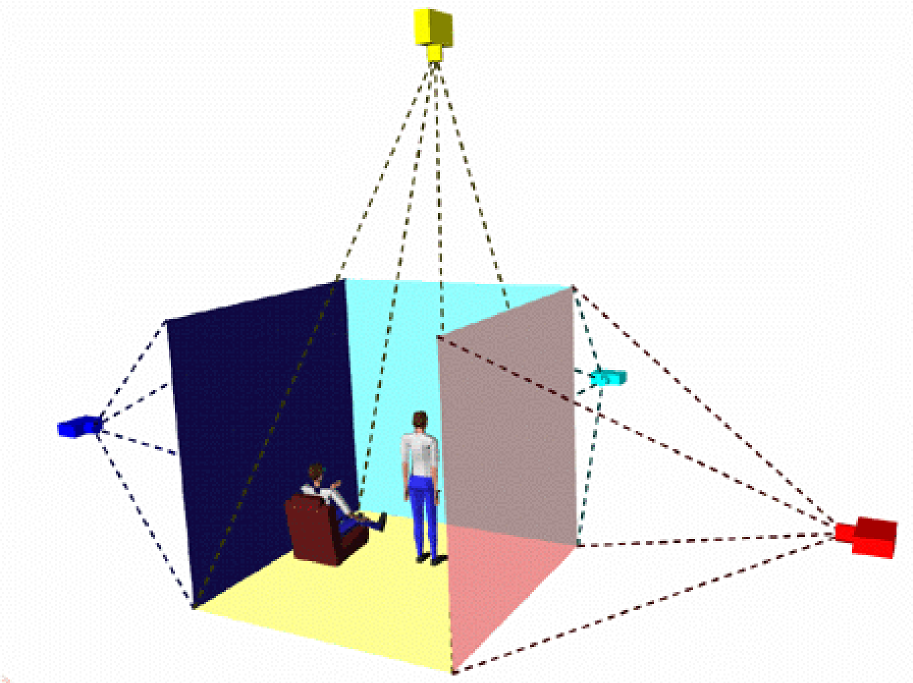

Figure 1: Cave Automatic Virtual Environment

3. Microsoft Kinect

This Microsoft Kinect consists of a horizontal bar connected to its base via a small motor. It allows the camera to make small movements up or down in order to adapt the perception of the camera according to the user`s position. The horizontal bar is the main element of Kinect technology. It contains a series of multi-microphones, an RGB camera, and finally a 3D depth sensor to improve the analysis of motion.

Figure 2: Microsoft Kinect

The Kinects are installed above the projection surface of the CAVE. This position is optimal for tracking; this position enables the Kinect cameras to detect the entire space obliquely from above the CAVE (see Figure 3).

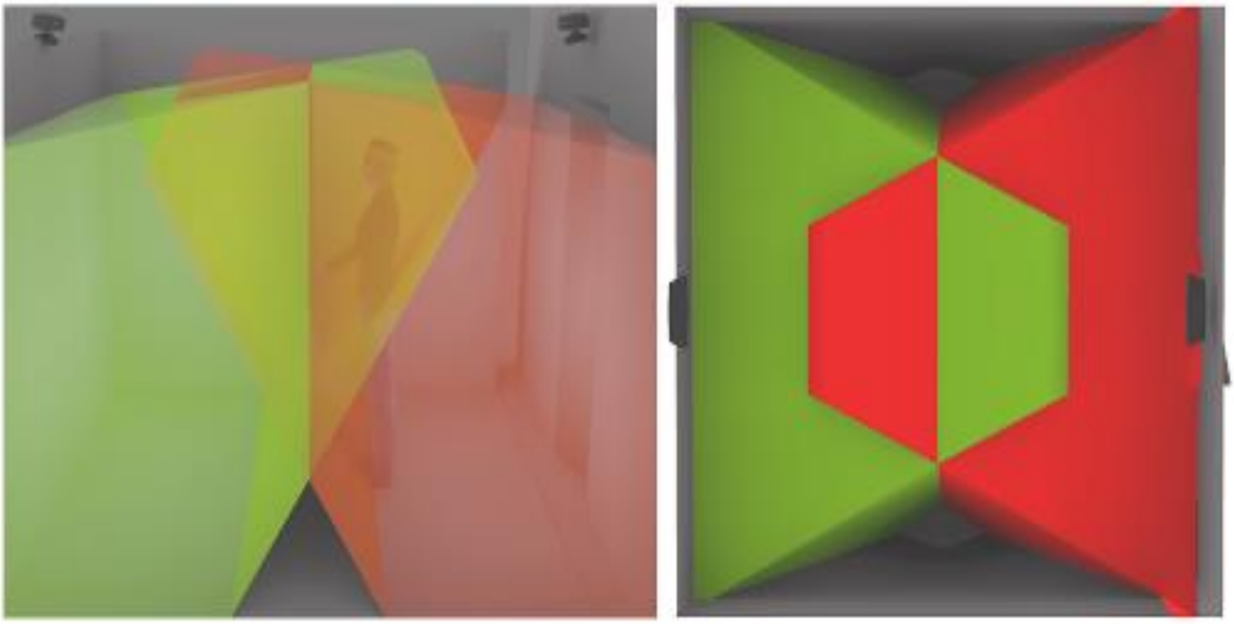

Figure 3: position of the Kinects in the CAVE

The two Microsoft Kinects do the tracking of the skeleton of the user via the SDK (Kinect-Software Development Kit) [2], the joints described in the next Figure are recognized by the SDK. These joints have three values corresponding to the Cartesian coordinates of the articulation point in the X, Y and Z coordinate system of Kinect. This information is stored in a «SkeletonPoint».

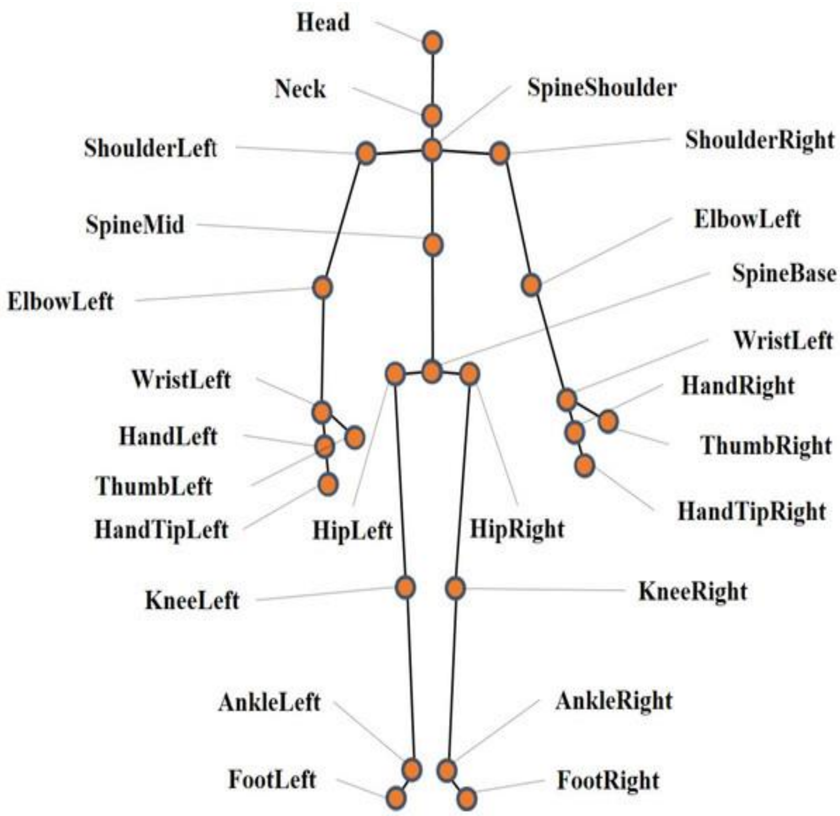

Figure 4: Kinect sdk recognised Joints

4. Virtual Reality Peripheral Network

The Virtual-Reality Peripheral Network (VRPN) system provides a device-independent and network-transparent interface to virtual-reality peripherals. VRPN's application of factoring by function and of layering in the context of devices produces an interface that is novel and powerful. VRPN also integrates a wide range of known advanced techniques into a publicly-available system. These techniques benefit both direct VRPN users and those who implement other applications that make use of VR peripherals [2].

The "VRPN-Server" program deals with the import of data such. Since this system already provides many solutions for hardware used in the Virtual Reality (head tracking, data gloves, wands, etc.) and the system is also expandable, the implementation was done via a native plugin for the engine.

5. MULTIKINECT

MultiKinect[1] is based on the use of multiple Microsoft Kinect sensors at a time in order to track users and detect gestures performed by them. We have developed an application that allows getting a skeletal representation of each user placed in front of the sensors. The program collects, processes and combines the data from each of the connected Kinect sensors, and makes it accessible through a VRPN server so that external applications may access.

The main reason of using multiple Kinect sensors is increase the accuracy of the final skeleton. Working with a single sensor, the resulting accuracy is very low because the field of view is too small. Increasing the number of sensors, we accomplish to increase the field of view of the system this allows getting more accurate final skeletons

II. HTW-CAVE current state analysis:

1. Hardware:

The CAVE of the HTW Berlin has a square plan of 3m edge length, it has a special polarization-preserving floor and back projection on three of the side walls. The projection onto these four sides takes two commercially available wide-angle projectors with a resolution of 1024 × 768 pixels.

For psychological reasons and because setting up a projector for the back wall requires a complicated mechanism, the back wall of the CAVE has no projection.

Two Microsoft Xbox 360 Kinects operating with a projected infrared pattern, each centered at 2.35m above the front screen and the back wall, track the user and capture his skeleton.

[1] https://github.com/mvm9289/multikinect

Another important component is the Nintendo WiiMote and it allows the user to interact with the virtual world.

2. Software:

In the software side, the data from the input devices have to get into the application from the input nodes and this must run parallel and synchronously on all rendering nodes. there is a cluster of ten nodes, eight of which are each responsible for the output of the projectors and two are responsible for connecting the input devices.

An input node takes the role of the master, which accepts the inputs and calculates the logic and physics of the applications. All changes to mobile objects are replicated for each frame to the running applications of the rendering nodes, thus enabling synchronous rendering.

The CAVE of the HTW Berlin has three independent VRPN servers for the WiiMote and DirectX controllers, as well as for the Kinect.

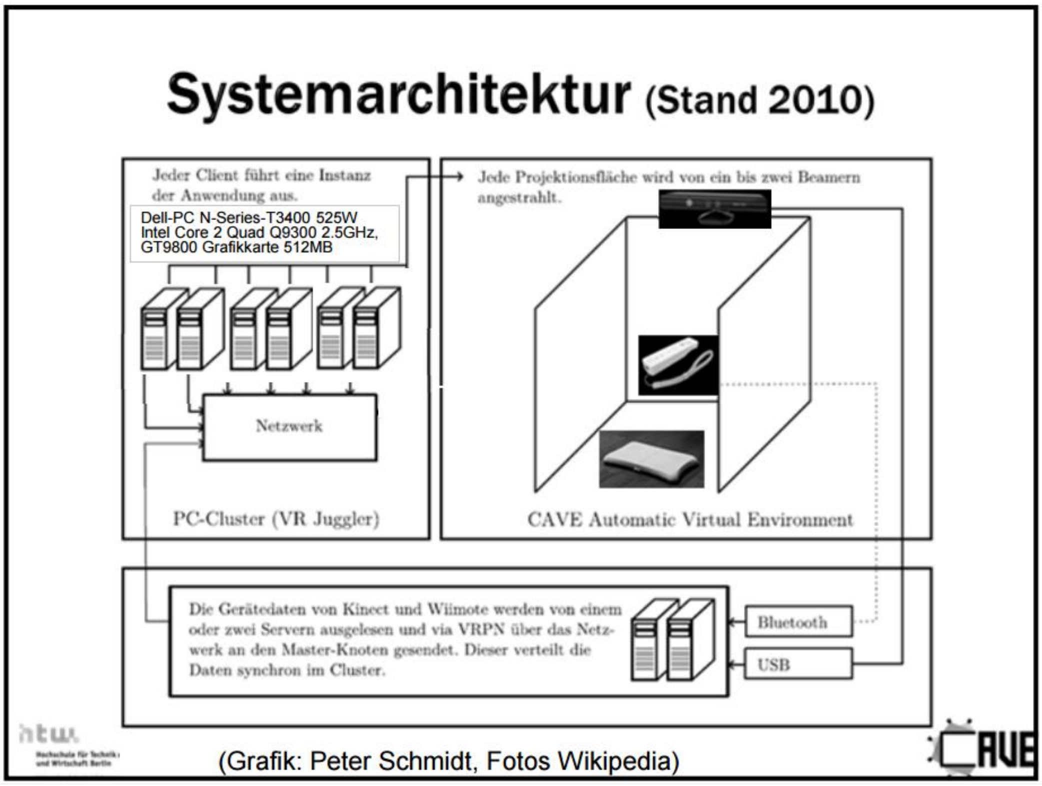

Figure 5: System architecture

3. Existing technology for the movement in the Cave:

Like many other Virtual Reality Interfaces, the CAVE suffers from applications that require movements, so currently a controller-based technique is implemented for moving around using the Nintendo Wiimote controller. a connection to a personal computer is possible due to Bluetooth-based communication. The input device must first be registered with the operating system as a so-called Human Interface Device (HID).

The entire scene can be rotated by pressing the corresponding button, to move left or right or to move to the front or back, it is irrelevant how the device is held.

The application of such a technique provides a high degree of control. By using a joystick, a directional change is also possible during the movement.

Figure 6: Nintendo Wiimote Controller

Since the position of the user in the CAVE is disregarded, this form of movement is not very immersive.

4. Free Controller Interaction for the movement in the Cave (Gesture interaction):

One of the solutions to interact with the CAVE system without the use of a Controller is to detect the user's gestures and use them to perform movements or actions within the virtual environment.

a) Tracking:

The term "tracking" describes all processing steps that are necessary for tracking moving objects. The aim is to depict the observed movement for technical use. One example is the skeletal tracking used in this work, in which striking points - so-called joints - are tracked by a person. These are usually hands, elbows, shoulders and head. Thus, a digital representation of the skeleton can be reconstructed. [5]

b) Gestures

A gesture is a movement of the body that contains information. Waving goodbye is a gesture. Pressing a key on a keyboard is not a gesture. All that matters is which key was pressed. [6]. You can also distinguish between two types of gestures. On the one hand, continuous gestures, in which the observed movement is connected with a state in the computer. For example, a pointer (cursor or pointer) can be moved over the screen. On the other hand, discrete gestures that are associated with an action in the computer and cause a change in the state.

c) Gestures recognition

In the detection of gestures, the observation data of the sensor system are analysed by various algorithms in order to extract and classify certain features.

You can distinguish between device- and camera-based detection. In the case of device-based recognition, the user carries the sensor system on the body. Examples of this are data gloves1 or the controllers of the Nintendo Wii2. In our case, the case of camera-based detection, the user is free of any peripherals. The gesture recognition is carried out on the basis of individual images or image sequences which are recorded by one or more cameras. One such recording device is the Microsoft Kinect used in this work.

III. A possible approach in part B:

The architecture of the proposed system includes four components: Kinect, CAVE, gesture recognition, and gesture database.

The goal of the system is to recognize a sequence of predefined movements performed within the cave and to identify the occurrence and duration of these movements. The recognition module extracts occurrences from the user performance and then assesses it against predefined movements and finally executes the action.

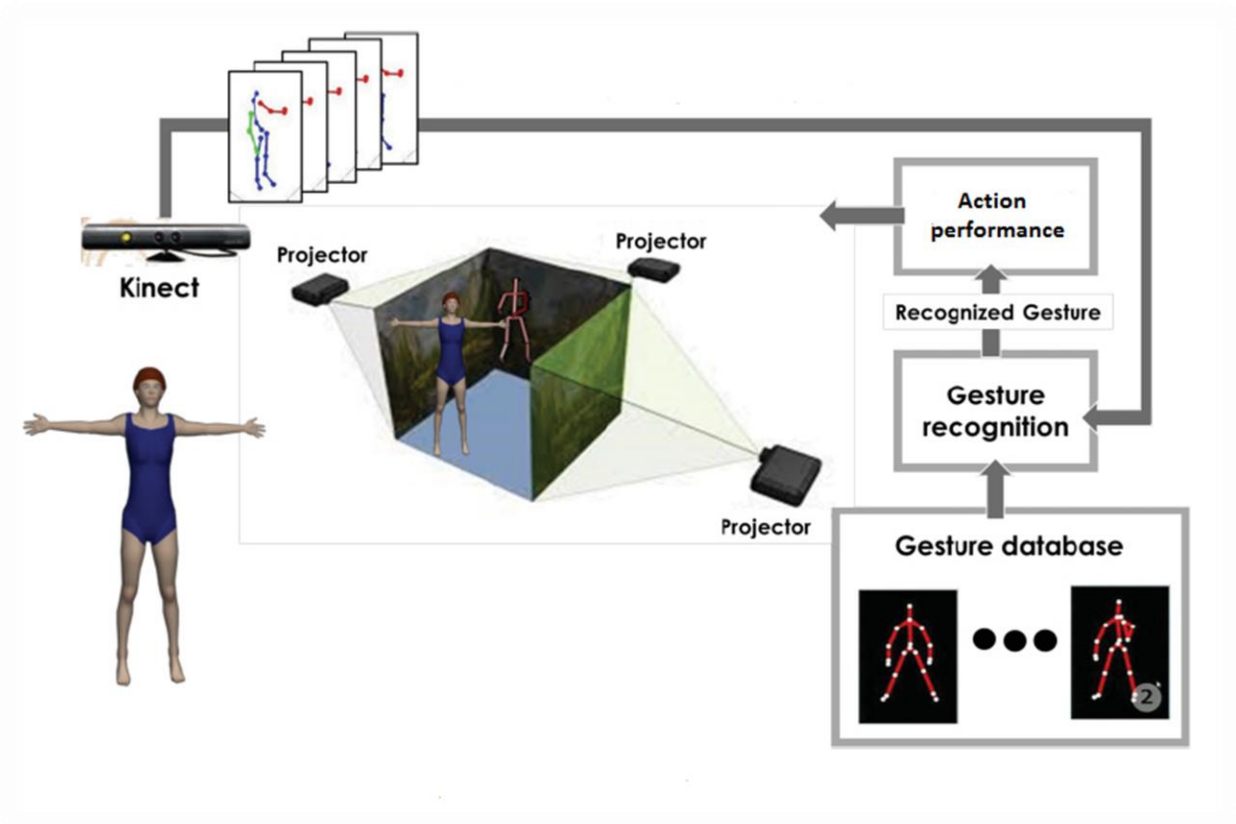

Figure 7: Proposed system architecture

References:

[1] Cruz-Neira, Carolina; Sandin, Daniel J.; DeFanti, Thomas A. ; Kenyon, Robert V. ; Hart, John C.: The CAVE: audio visual experience automatic virtual environment. In: *Commun. ACM* 35 (1992), Juni, Nr. 6, 64–72. http://dx.doi.org/10.1145/129888.129892. – DOI 10.1145/129888.129892. – ISSN 0001–0782

[2] Microsoft: *Programming Guide Natural User Interface: Skeletal Tracking. http://msdn.microsoft.com/en-us/library/hh973074.aspx*.

[3] Kurtenbach, G. & Hulteen, E. (1990). Gestures in Human-Computer Communications. In B. Laurel (Ed.) The Art of Human Computer Interface Design. Addison-Wesley, 309-317.

[4] Kurtenbach, G. & Hulteen, E. (1990). Gestures in Human-Computer Communications. In B. Laurel (Ed.) The Art of Human Computer Interface Design. Addison-Wesley, 292-307.

[5] wiki: https://en.wikipedia.org/wiki/Video_tracking

[6] Edwards, L. 2002. A natural history of mathematical gesture. Research report presented at "Mathematics and Gesture" Symposium, American Educational Research Association Annual Meeting, Chicago.