

Fire fighting and related simulations in a CAVE using off-the-shelf hardware and software

Frank Poschner

Department of Computer Engineering
Faculty of Electrical Engineering and Computer Science
University of Kassel, Germany

Abstract

The growing interest in using Serious Games in education has also reached the field of fire brigade training. By the use of simulations, firefighters can train in virtual worlds on the computer and learn about processes and prepare real fire drills. It is for example possible to extinguish a virtual fire or to set up a water supply in such simulations. Played on a common computer and with input devices like mouse and keyboard, the games normally do not offer a very realistic impression on the user. For an immersive experience, those virtual training games must be used in Virtual Reality environments like a Cave Automatic Virtual Environment (CAVE), which is usually associated with high costs. In this paper the setup of a CAVE used for simulations in the field of fire brigade training will be presented as well as the use of input devices and their coupling with real firefighting equipment. Since almost exclusively low-cost and commercially available elements such as common projectors, the Microsoft Kinect and the Nintendo Wii controllers were used, at the same time an approach for bringing simulations like these into VR environments without large costs is demonstrated. As the presented simulations are developed with a free but powerful Game Engine, a wide range of possibilities for the development of training games for a CAVE environment is given.

Categories and Subject Descriptors (according to ACM CCS): I.3.1 [Computer Graphics]: Hardware Architecture—Input devices, I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism —Virtual reality

1. Introduction

A fire has broken out in a garbage can on the street. How should a civilian behave now, what would a firefighter do? The matching behavior of the latter are taught and trained in the exercises of fire brigade training.

The use of games as a basis for training and learning can be basically described with the expression of "Serious Games" [MC05]. Such games exist in many different types, special attention is recently given to training simulations on the computer. They are widely used in different educational sectors like health and safety, but even in political education [Pre01].

The use of such games is also of interest in the field of fire brigade training and for teaching in preventive fire protection. The effective support of education through preparation, follow-up and support of practical exercises is tried to be achieved here. Furthermore, the objective of saving time of

presence by giving the trainees the opportunity to be trained in the practical processes and their rules by simulations on a personal computer at home or even on mobile devices is pursued. Expenses in time, money and organisation can be saved. Such Serious Games can be of very different types to the point of those consisting of complex, virtual worlds in which avatars of the player can be independently moving in an environment to solve problems and maybe even being connected to others via network [ES14]. Simulations like these belong to the area of Virtual Reality applications [Rhe91]. The player should be able to immerse himself into the virtual world, also to support the learning effect. It is known that the creation of immersion for a player goes hand in hand with a realistic representation of the game scene on matching displays and the use of ergonomically sensible and appearingly realistic input devices. It would therefore be most useful for the player of a simulation in the field of fire brigade training, when he gets the impression of a game

scene in a realistic size and could use realistic equipment. A solution would be to use a CAVE (Cave Automatic Virtual Environment) and input devices that create the impression of real equipment to the user.

Operating such a CAVE means to set up a well cooperating configuration of hardware and software to link multiple projection surfaces or screens to a virtual environment for an immersive impression of the user. Originally, the operation of a CAVE required specially developed hardware and software. In particular, the operation of parallel computer systems, the use of appropriate software and the connection of special input devices was often necessary. This special equipment usually made it a costly affair to operate a CAVE and would therefore not meet the needs of the fire brigades. In Chapter 2 of the presented work the operation of CAVEs will be discussed in general as well as related works that benefit from the progressive development in the field of gaming technology by using commercial products in a CAVE. In the further chapters a solution for the development of a CAVE with 3 walls is presented as well as simulators from the field of fire brigade training and preventive fire protection which were developed for the CAVE with a freely available game engine. Furthermore, the use of input and output devices that are meant to create the impression of realism to the player is described. For example, the Microsoft Kinect and the Nintendo Wii controller, which was coupled with a fire extinguisher, were used. Almost all elements presented here are common products in the field of game industry and game development, so that this work also represents a proposal for the low-cost design of a VR environment.

2. Related work

The concept of a CAVE was first presented in 1992 by Carolina Cruz-Neira et al. [CNSD*92] as an environment for virtual reality applications. Based on this proposed approach many CAVEs were built, which were equipped with special hardware and software configurations. During that time, the hardware required for a CAVE such as projectors, the computers and graphics hardware for the processing of the images as well as the matching input devices was expensive. Magnetic or optical tracking systems were used for recognizing the user's movement while the interaction with virtual objects in the scene could for example be realized by the use of data gloves. Special software such as VRJuggler [VR14] was developed to support parallel rendering, image processing and the use and management of input devices. Due to the enormous development in the field of computer and graphics hardware, computer games and game consoles as well as input devices for gaming there are new ways for the use of these technologies by including them into CAVE systems. Jung et al. [JKS11] use the Microsoft Kinect, which is originally an optical input device with depth sensor for the Microsoft Xbox console, or even two of them [JS12] to track the motions of the user for navigating a virtual scene by gestures and even for grabbing objects. For the process

of grabbing itself a Nintendo Wii controller is used. Settgast et al. also use the Kinect for navigating in an "affordable" CAVE-like environment and compare its use for navigation to that of a joystick [SLBF12]. Furthermore a work for the integration of the Wii controller into VR applications is presented, in which the Wii is used for different purposes like playing drums or recognizing the user's gestures [HPL07]. While gestures can be used for navigation like raising an arm for stepping forward as shown by Jung et al., this might not be satisfying from an ergonomic point of view. One attempt for a more realistic way of using input devices for walking and running that uses common games hardware is presented by Hilsendeger et al. [HBTF09]. The Nintendo Wii Balance Board which was originally built for fitness applications is used here. The player is able to navigate a scene by standing and stepping on the buttons of the board. There are new attempts in this field of navigation like the Virtuix Omni [Vir14], which is again specially designed VR hardware and therefore expensive or not yet available. As one goal of our work was to build CAVE simulators with low cost components, we used almost common hardware only. In a similar project a CAVE was built and connected to planning simulations in the field of factory planning [Ins13]. This work presents solutions for firefighting and related simulations in a CAVE. There are many simulations in this area, which are typically developed for personal computers or mobile devices. One of those is a simulation which is used as a basis for our work and has been developed in the project "KATIE". It is a network game in which the player can cooperate with others in a virtual environment. He can also use virtual equipment of fire brigades for certain operation purposes [Pos13].

3. CAVE setup

In this chapter the setup of the CAVE used for fire fighting simulations is described. This covers a short description of the basic hardware components as well as the idea and the steps of how to create the CAVE view.

3.1. Hardware components

The hardware components consist of two different parts: The CAVE frame and the components used for processing the CAVE images.

3.1.1. The frame

The most basic elements of the CAVE are their walls and screens. The CAVE has three walls which are ordered rectangular and have a size of 2.5m x 2.5m each. The frame of each wall was built out of wooden pillars between which canvas for back projection was stretched, the left part of figure 1 shows the CAVE out of action. As the building process itself is not of interest at this point it can be reviewed in the

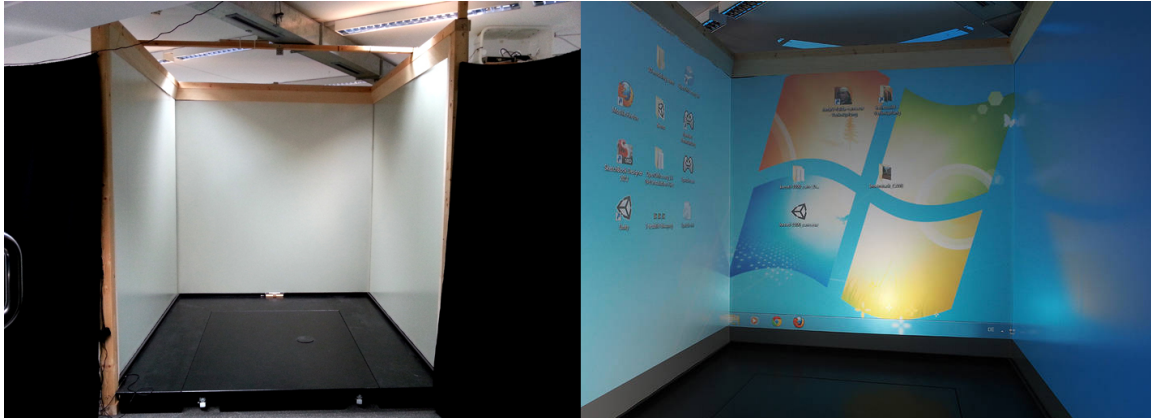


Figure 1: Left: CAVE turned off. Right: CAVE showing a desktop view.

thesis under which it was developed [Wol02]. All the described solutions could also be used on a single wall or a projection screen.

3.1.2. Components for image processing

As already pointed out only standard components are used for operating the CAVE. In former times typically a lot of computers had to be connected for a good performance in a CAVE. For this CAVE only one computer is used, which is a normal PC with an Intel i7 processor, 12GB of RAM and an Asus P6T S1366 mainboard. Three graphics boards of the type NVIDIA GeForce GTX 285 with average performance are used.

Three projectors of the brand Optoma are used for the image projection. The fact that the targeted image resolution can be represented in 120Hz is important here. In our case this resolution is 1024 x 768. As the CAVE walls are quadratic we would need a resolution that has the same width as height. This is normally not supported by the projectors available and so 256 pixels of each projection width are cut off by considering this in our simulation. The projectors need to be placed in a certain distance to the CAVE walls, in our case this is 2,965 meters. In case the room of the CAVE is not big enough, mirrors for reflecting the images can be used [Wol02].

For the stereoscopic rendering an active stereo system with shutter technology developed by NVIDIA is used, which is described in the next section.

In order to bring 3D sound into the CAVE an active 5.1 speaker system from "Logitech" is used, which is connected to the built-in sound card of the computer mainboard.

3.2. CAVE view and stereoscopic rendering

To create a CAVE view, as many images as the CAVE has walls have to be projected in a way that the images are seamlessly combined. For this reason we combine our three

graphics boards as an SLI combination connecting each projector to one card. NVIDIA offers the matching technology for their cards in the graphics driver, which is named "NVIDIA surround". While this works with monitor displays it is also possible to connect projectors. The configuration is a simple process in the supplied system control. The right side of figure 1 shows the CAVE with working NVIDIA surround showing a Windows desktop.

The next step is to establish stereoscopic rendering in the CAVE. As already mentioned, the technology of active stereo with shutter glasses is used here. NVIDIA offers a USB infrared emitter which has to be connected to the computer and sends shutter signals to the matching shutter glasses [NVI14]. This system can only be used with NVIDIA graphics boards and represents a simple and low-cost solution if the user doesn't want to implement the stereoscopic rendering himself. The NVIDIA driver recognizes the start of a 3D application like a game and renders the images ready for shuttering. There are parameters that can be set like the depth of the 3D effect. As all these components work together well in the CAVE, serious games can be developed or adapted for the use in this environment, which is described in the next chapter.

4. Developing CAVE simulations with a free available game engine

This chapter describes the use of a game engine for creating simulations for the presented CAVE system. Since the generation of simulations with variable and rich content was always a difficult task in conjunction with CAVEs and their specialized software, it was our goal to try using today's gaming technology in this regard. The approach was to combine simulations developed with the game engine Unity [Uni14] with the setup of a CAVE like it is described in chapter 3.

Unity is a free game engine with a wide range of functions

for the development of games and simulations. The basic idea was to be able to use Unity simulations that have been developed for a variety of devices in the CAVE. This chapter describes the necessary adjustments for the CAVE view. Necessary adjustments for the use of special input devices are discussed in Chapter 5. The CAVE installation includes

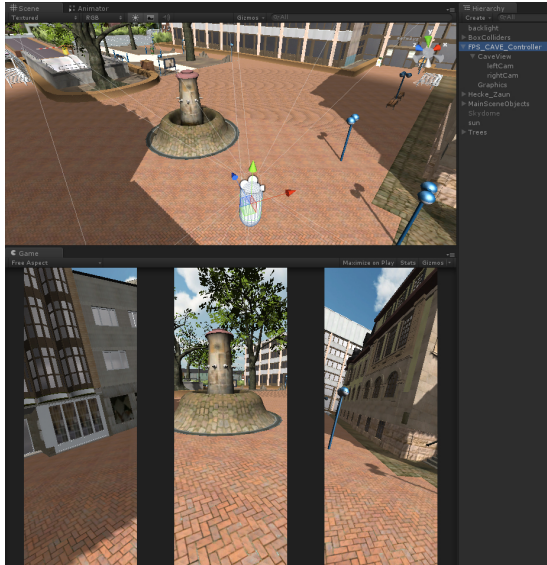


Figure 2: Development view in Unity for CAVE simulations.

three square screens with rear projection. A game scene must be represented accordingly by three images in a suitable arrangement. For this purpose, three virtual cameras must be used in the game scene and be structured to fit and work with the appropriate viewport. Therefore a package of Unity elements is created, which represents a controller to the corresponding cameras. This can be used for navigating a Unity game scene. If other controllers or representatives of the user have to be developed for the CAVE, this controller must be adjusted accordingly. The camera settings and positions can then be applied.

Figure 2 shows a picture of the First Person Controller in a game scene and the arrangement of the cameras for the CAVE view. In the Hierarchy view the hierarchy of the used components can be seen. The GameObject "FPS_Cave_Controller" at the highest level is responsible for controlling the FPS object, this must be adapted for an adaptation of the controller to other needs. The attached GameObject "CaveView" has a script "CAVE", which is responsible for setting the camera values at runtime. The attached GameObjects "leftCam" and "rightCam" are also generated at runtime, which can be seen in the screenshot. "CaveView" itself is responsible for the camera of the center wall.

Regarding the arrangement of the CAVE screens at right angles and taking into account that seamless image transitions

between the screens have to be created, the field of view of the cameras has to be adjusted. It must also be noted that supernatants of the images would arise on the lateral borders of the screens, because of their square format and the restrictions in the resolution settings of the projectors. Those can only provide images in either 16:9 or 4:3 format. Since 4:3 format will be used in the CAVE, the camera images have to be cropped by using black stripes to prevent picture supernatants between the images when using the full height of the screens. Figure 2 also shows the game view of the scene at runtime using the described stripes.

The camera producing the image of the left wall must be rotated by -90 degrees about the y-axis to that of the center wall, while the camera for the right wall must be rotated by 90 degrees to the camera of the center wall. In figure 3 the corresponding settings for the Unity camera that captures the image for the right CAVE wall are shown. While the settings of the clipping planes can be adjusted to the different simulations, the field of view should be 90 degrees for each camera. The height of the Normalized View Port Rect should be 1 for all cameras, the width of the viewport should be 25 percent each, which was determined in tests in the CAVE but can also be calculated. To generate the stripes described earlier, the x-component of the viewport, which describes the starting point of the camera picture from the left to the right border of the screen, depends on the respective camera position: There must be black stripes to the left and the right of each wall. This results in six similar stripes, which means a width of $(100 - (3 * 25))/6$ for each of the stripes. The

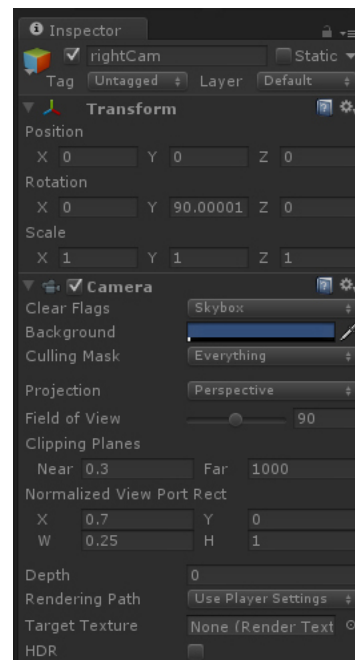


Figure 3: Exemplary settings for the Unity camera that produces the image for the right CAVE wall.

camera of the left wall starts with the width of one of these stripes as offset in the x-component of its viewport, while the center camera starts with an offset of 0.25 plus three times the width of a stripe. Figure 3 shows the right cam starting at 0.70 due to some rounding, which is compensated by the position of the projector.

The Unity Input Manager should be used to operate the FPS_CAVE_Controllers by Mouse, Keyboard, Joystick and XBOX360 controller.

Sound in the CAVE results directly from the use of sound in Unity, the settings of the Windows sound driver and the sound hardware. Unity offers a range of settings for audio sources, surround sound is possible and must be set in Unity, too.

To link CAVE simulations via network with other applications, for example to develop a multi-player mode, the Unity network functions can be used. For this purpose the CAVE computer must be connected with an appropriate network, as it is common for PCs with a Windows operating system. Applications for the CAVE must be exported as a PC standalone application from Unity3D with the projector's resolution while the width is taken times 3. In our case the resolution is 3072 x 768.

5. Input devices

An important reason for using Unity simulations in a CAVE is the ability to use various commercial but low-cost input devices in simulations. This chapter describes some of the connected devices and how they are used. Of course standard devices like mouse, keyboard and common joysticks can be used, especially when connected wireless. Although this may be not of great interest, it should still be mentioned.

5.1. Using Microsoft Kinect for head-tracking

Detecting the position of the player in a CAVE can give developers the possibility of directly reacting on the human player's physical behavior in a VR application and connect him directly to the virtual world. Especially the detection of the position of his head has an important meaning, because the eye position of people can be important for the proper recognition of 3D effects. In Unity the player looks into the virtual world through a camera. If he bends down, the camera can also be moved down in the same amount like he moved his head. Thus, the user will get a more realistic impression of the scene.

Originally expensive special hardware was needed for this head-tracking. This often had to be fixed to the head of the player. An example is the "Flock of birds", a magnetic position detection system [Fif14]. Since the release of the Microsoft Kinect, a camera system that has a depth sensor and can be connected to a PC, the detection of movements of a player can also be done with this system. It is connected via USB and a matching SDK to use the system with Unity is

provided. The advantages of using the Kinect are the elimination of body-worn parts for head-tracking and the possibility of tracking and distinguishing between different body parts, not just the head.

In our CAVE the Kinect is mounted on the open side of the CAVE on the opposite of the center wall at a height of 2.80 m, by that it is behind the usual sight of the CAVE user. Since the Kinect has a specified minimum distance of 1.80 m to the tracked person and because of negative results in tests with lens attachments to reduce this distance, a fixture at another location was out of the question because of the height of the CAVE frame. The distance to the users would be too short and the angle at which the camera focuses the users would be too large to grant a reliable recognition of the user's head. By the actual mounting the tracking of movements, carried out by the player towards the center screen in front of his body, is not possible though. In general, the sense of using the Kinect is mainly directed to head-tracking, although body-tracking is also possible. Figure 4 shows the current position of the Kinect in the CAVE.

The Kinect is connected to the CAVE computer via USB.



Figure 4: The Kinect sensor mounted on the rear end of the CAVE.

The following problem may appear with the use of the Kinect: If many USB devices are connected to the CAVE computer, it may lead to utilization of the computer's USB bandwidth. The utilization of the Kinect is classified as very high. Symptomatic of this is a high latency in the motion detection by the Kinect, which results in jittering in the movements of the tracked player. Since the CAVE projection is not supported by external light, but only the light from the projector itself is used, there may be an insufficient illumination of the CAVE in simulations with dark scenes. Tracking errors may occur due to the Kinect then, and the player's position in the simulation might not be affected any more. On Windows operating systems the Kinect can be used by installing the Microsoft Kinect SDK which is used here, too. To use the Kinect in Unity, a matching software package must be integrated. A package of "Carnegie Mellon University" has been used successfully here [Car14].

5.2. Nintendo Wii controller and real equipment

The Nintendo Wii Controller or Wii Remote is a wireless input device, which was originally developed for the Nintendo Wii console. It has a number of buttons, a directional

pad and acceleration sensors, which make it possible to detect the movement of the player who holds the controller. Furthermore, a goal can be targeted, which is done via an infrared source to be sighted. The controller can be extended with the so-called Nunchuck controller, which has an analog stick, two buttons and also acceleration sensors. Since the Wii controller can be connected to a PC, it is also suitable as an input device for PC-based games. It is used as an input device with the extension of the Nunchuck controller via bluetooth connection in the CAVE. In order to use the Wii controller with Unity, a suitable framework must be used. The free framework UniWii has been used here.

The advantage of using the Wii controllers in a CAVE is the possibility of attaching it to real equipment. It can, for example, be coupled to a fire extinguisher. Its acceleration sensor and the possibility of targeting infrared sources allow the use of sensor data in Unity simulations. For example the user can target a virtual fire in a simulation by using the hose of the fire extinguisher. Subsequently a button of the controller, which is attached to the appropriate place of the fire extinguisher is pressed and a virtual jet of water can be released, which triggers the deletion of the virtual fire. Another possibility would be the attachment to an axe to simulate the hammering of a door or to use it with a jet pipe for firefighting. Figure 5 shows the coupling of the Wii controller to a real but empty fire extinguisher. An ergonomically sensible position was determined first and then the controller was tied to the extinguisher by using cable ties.

Since the Wii controller also has an analog stick and different buttons, it can be used for navigating a simulation scene and thus make the use of further controllers unnecessary.



Figure 5: Using the Wii controller with a fire extinguisher.

5.3. A Gamepad for navigation and handling

Gamepads can also be used as input devices in the CAVE. They can be integrated by using the Unity Input Manager and have their advantages in the easy and ergonomic operation, particularly for the rapid handling of equipment or to navigate through scenes. In the case of the presented CAVE an Xbox360 controller is used, which can be operated via the standard drivers in Windows. The mentioned FPS_CAVE_Controller can be controlled intuitively using the joysticks and buttons on the gamepad.

6. Simulations

The various input devices described in Chapter 5 were used and tested in various simulations for the training of fire brigades and fire protection, which is described in this chapter.

6.1. Turntable ladder vehicle tutorial

To use a turntable ladder vehicle requires an extensive knowledge. In order to explain the basics of controlling the ladder a Serious Game was developed. It allows the user to

train in the CAVE and includes a user guide with explanations. It begins with explanations to the use of the vehicle and ends with the rescue of a man from the roof of a house. The game is controlled by the user via gamepad to interact with the car controls, but also head-tracking using the Kinect is implemented. The users get a realistic impression of the virtual world, because the environment reacts on their behavior. The games is played in a first-person mode, but also a mode for 3rd-person operation was implemented. In the 3rd person mode the movements of the avatar that represents the player are tracked, too. This was only used as a test of the Kinect, because the appearance of the player's avatar in the simulation is not necessary: The player should feel as a part of the virtual world and not see his body virtually represented. Figure 6 shows the simulation in the 3rd person view with the avatar in front of the ladder vehicle. It is shown in the Unity development view while the user raises his arm. The Kinect works well in this simulation, sometimes there can be a few jittering effects due to the light conditions. The basis for this simulation were extracted from the literature and discussions with firefighters. It is planned to make it available for testing by experts from the fire brigades soon.

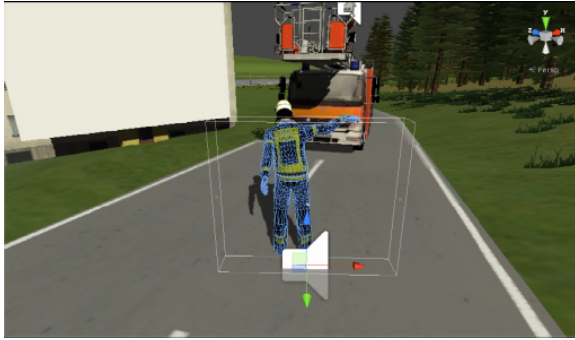


Figure 6: User is tracked by a Kinect in the simulation of a turntable ladder vehicle.

6.2. Fire in a residential building

In this simulation, the Nintendo Wii controller is used. At the start the player is located in a bathroom of a residential building and can navigate in first-person mode with the joysticks and buttons of the Wii controller, that is mounted to the fire extinguisher controller. He can open doors and needs to find a fire that has broken out somewhere in the building. The game is equipped with instructions for the player. He has to take a fire extinguisher from the basement of the virtual house and extinguish the fire with it by using the extinguisher like he would do in the real world. A particle system simulates the extinguishing media and interacts with the virtual fire if used correctly: The fire extinguisher can be operated by targeting the corresponding IR-emitter with the Wii controller. This simulation shows small accuracy problems when targeting the IR emitters, but generally the integration of fire extinguishers and Wii controller and the connection to the Unity simulation work well.

6.3. Using a turntable ladder vehicle and deletion of fires in an industrial area

In this simulation, the player starts in an industrial area, in which a turntable ladder vehicle and a burning fire tank can be found. At the beginning the player's movements are controlled by the Xbox360 controller. The user has to find a fire extinguisher and take it. When he takes the virtual fire extinguisher the controls change to the Wii controller, mounted to the real fire extinguisher: The analog joystick of the Nunchuck can be used for walking while also the infrared sensor can be used to change direction. Also the player can change his line of sight by using the Wii controller to point in a certain direction so he can target a fire. The game contains a tutorial: The fire at the tank truck must be deleted first, then a turntable ladder vehicle must be operated correctly. The user can drive the ladder to a fire on the roof of a hall, being in the basket of the ladder. That fire has to be extinguished, too. Figure 7 shows a user in the CAVE extinguishing the fire at

the tank truck using the fire extinguisher in combination with the Wii controllers. The simulations where the Wii controller is used were tested by a fire safety expert of the fire brigades Kassel and classified as appropriate for use in preventive fire protection. A revision and expansion of the content to different scenarios is in development. It can be noted that the

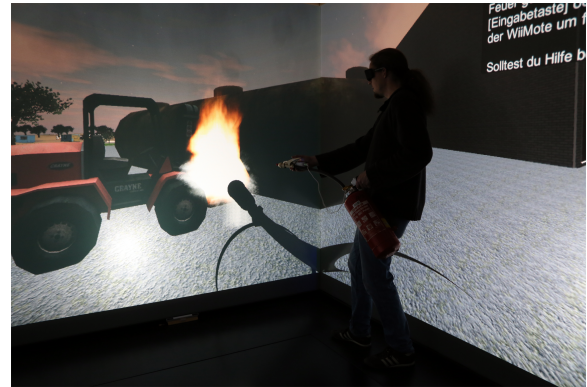


Figure 7: Extinguishing a virtual fire with real equipment.

use of the Wii controllers for navigation and targeting a fire has been implemented successfully and works well in this simulation .

6.4. Using virtual equipment and setting up a water supply

As already pointed out there exists a simulation that was used as a basis for creating applications in the CAVE with the presented setup. By using the Xbox360 controller virtual equipment of fire brigades can be used for certain operation purposes here, like setting up a water supply. The simulation itself is described in [Pos13]. The contents of the simulation was developed in cooperation with experts from the County Fire Services Association Fulda. A PC version which is controlled via mouse and keyboard was evaluated and revised by intensive tests with firefighters of voluntary fire brigades. However, the CAVE version shows ergonomic problems being controlled by the gamepad, especially if the user is inexperienced. While not all operable objects of the simulation can be represented by real equipment, the usability of the presented input devices comes to a limit here and a more natural interface has to be found. In figure 8 two people can be seen using the simulation in the CAVE.

7. Conclusions and future work

In this paper we have presented the work of developing simulations for the field of fire brigade training in a CAVE, using common hardware and software like the Nintendo Wii game controller and a free game engine. With the game engine



Figure 8: A simulator for fire brigade training.

Unity it is possible to create simulations that can be run using various input devices in the CAVE. With the solution shown, it is also possible to generate stereoscopic projections on 3 CAVE walls for the applications. Furthermore, equipment like a fire extinguisher is coupled with the controllers and used in a way that real equipment becomes an input device. As the hardware and software used here is relatively inexpensive, this work can serve as a template to build affordable CAVE systems. With this approach also VR systems with a single projection are conceivable. Different kinds of simulations have been presented to prove that the range of use of this system is wide.

In the future simulations could be developed that include all the input devices in a sensible way as the integration of all the presented elements in one simulation has not yet been accomplished. This is mainly for the reason that problems such as the lack of USB bandwidth must be solved for the simultaneous use of all devices. Furthermore, a solution for the problem of moving in a way that simulates real movement should be developed in the CAVE. While head-tracking is already possible by using the Microsoft Kinect, to track walking and running with it would lead the user out of the range of the CAVE and so at present the Xbox360 controller or the Nintendo Wii controller are still used for navigation. It is also planned to develop input devices with sensors based on components like Arduino products [Ard14] to use real equipment of for example fire brigades as input devices. Much better ergonomics could be the result as the small sensors could be integrated into the equipment much more inconspicuous and at the same time provide results that are more precise.

References

- [Ard14] ARDUINO: Arduino - products, 2014. <http://arduino.cc/en/Main/Products> (28.04.2014). 8
- [Car14] CARNEGIE MELLON UNIVERSITY: Microsoft kinect Ü microsoft sdk, 2014. <http://wiki.etc.cmu.edu/>

- unity3d.com/index.php/Microsoft_Kinect_-_Microsoft_SDK (26.04.2014). 5
- [CNSD*92] CRUZ-NEIRA C., SANDIN D. J., DEFANTI T. A., KENYON R. V., HART J. C.: The cave: Audio visual experience automatic virtual environment. *Commun. ACM* 35, 6 (June 1992), 64–72. 2
- [ES14] E-SEMBLE: Xvr in general, 2014. http://www.e-semble.com/en/Products/XVR/In_general/ (27.04.2014).
- [Fif14] FIFTH DIMENSION TECHNOLOGIES: Ascension flock of birds, 2014. <http://www.5dt.com/products/pfob.html> (26.04.2014). 5
- [HBTF09] HILSENDEGER A., BRANDAUER S., TOLKSDORF J., FRÖHLICH C.: Navigation in virtual reality with the wii balance board. 6. Workshop der GI-Fachgruppe VR/AR, Aachen, Germany: Shaker Verlag GmbH. 2
- [HPL07] HAMMERL S., PREUSS T., LATOSCHIK M. E.: Wi-inc - wii network control - einsatz des wii-controllers für vranwendungen. *Virtuelle und Erweiterte Realität*, 4. Workshop of the GI VR & AR special interest group, pp. 141–148. 2
- [Ins13] INSTITUT FÜR PRODUKTIONSMANAGEMENT UND LOGISTIK: Virtuelle fabrikplanung in der 3d-cave (vifa 3d), 2013. <http://www.i-p-l.de/leistungen/cave> (28.04.2014). 2
- [JKS11] JUNG T., KROHN S., SCHMIDT P.: Ein natural user interface zur interaktion in einem cave automatic virtual environment basierend auf optischem tracking. *Workshop 3D-NordOst* (2011), 93–102. 2
- [JS12] JUNG T., SIMON F.: Interaktion in einem cave automatic virtual environment unter verwendung mehrerer tiefsensorkameras. *3D-NordOst* (2012), 199–208. 2
- [MC05] MICHAEL D. R., CHEN S. L.: *Serious Games: Games That Educate, Train, and Inform*. Muska & Lipman/Premier-Trade, 2005. 1
- [NVI14] NVIDIA: Nvidia 3d vision Ü the best stereoscopic 3d gaming experience for pcs, 2014. <http://www.nvidia.co.uk/object/3d-vision-main-uk.html> (27.04.2014). 3
- [Pos13] .. Eine virtuelle Multi-User Trainingsumgebung für die Feuerwehrausbildung (2013), Fraunhofer Verlag. 2, 7
- [Pre01] PRENSKY M.: *Digital Game-Based Learning*. McGraw-Hill, 2001. 1
- [Rhe91] RHEINGOLD H.: *Virtual reality*. Summit Books, 1991. 1
- [SLBF12] SETTGAST V., LANCELLE M., BAUER D., FELLNER D. W.: Hands-free navigation in immersive environments for the evaluation of the effectiveness of indoor navigation systems. In *VR/AR* (2012), Geiger C., Herder J., Vierjahn T., (Eds.), Shaker, pp. 107–118. 2
- [Uni14] UNITY TECHNOLOGIES: Unity, 2014. <http://www.unity3d.com> (28.04.2014). 3
- [Vir14] VIRTUIX: Virtuix omni, 2014. <http://www.virtuix.com/> (28.04.2014). 2
- [VR14] VR JUGGLER: vrjuggler - vr juggler provides a cross-platform virtual reality application development framework, 2014. <http://code.google.com/p/vrjuggler/> (26.04.2014). 2
- [Wol02] WOLF M.: *Planung und Aufbau einer immersiven Multiprojektionsumgebung*. Master's thesis, University of Kassel, 2002. 3