Using Virtual Reality in Education

Martin Němec, Radoslav Fasuga, Jan Trubač, Jan Kratochvíl
VSB-TU Ostrava/ Department of computer science, Ostrava, Czech Republic
martin.nemec@vsb.cz, radoslav.fasuga@vsb.cz, jan.trubac@vsb.cz, jan.kratochvil@vsb.cz

Abstract – The main purpose of this article is to describe and summarize experiments with creating training applications using virtual reality devices. The arrival of new and more advanced virtual reality headsets opens up more opportunities and applications. This article deals with the current state of virtual reality and its application in education. The first chapter describes basic information about virtual reality. Following chapters give a detail about some experimental applications that were developed for the virtual reality, and lastly, there are mentioned the benefits of those applications and the basic user reactions.

I. Introduction

With the increasing capabilities of modern computer hardware, more applications of these technologies are available in the field of education. Among these modern and fast developing technologies belongs virtual reality. This new trend demonstrates the enormous interest of large companies engaged in development and support. These include Google, Microsoft, HTC, etc. The interest in virtual reality is also seen in the game development when every year there are more and more titles which support virtual reality headsets.

The main purpose of this article is to describe and summarize experiments with educational applications for virtual reality, which we developed for testing and demonstration purposes of modern, widely accessible headsets.

II. Virtual Reality

First attempts at simulating complex systems came with the first 8bit computers, where users could simulate various systems through games. One such example is a game called Scram [1]. Scram (Fig. 1) is a game designed by Chris Crawford for the Atari 800 and released by Atari in 1980. Written in Atari BASIC, Scram utilized differential equations to simulate reactor behavior. In the game, the player controlled the valves and switches of a nuclear reactor directly with the joystick.

Attempts at simulating complex systems go back to the first 8bit computers, sometimes via the game media.

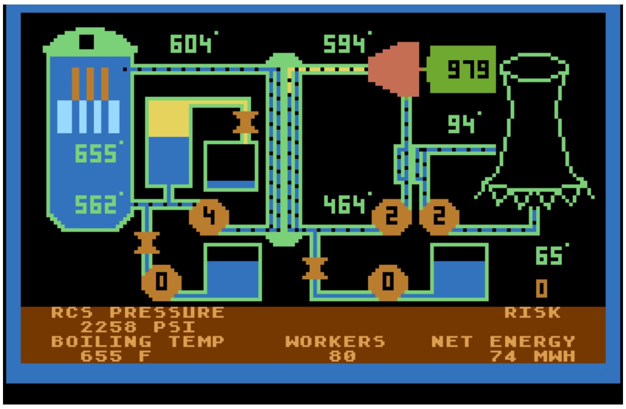

Figure 1. Game Scram which simulate reactor behavior [1]

The computer technology has advanced quite formidably and one interesting technology is the usage of virtual reality. With the help of virtual reality [2], we can simulate an illusion of real or fictional worlds, most commonly using special virtual reality headset that is worn on the head. In the headset, the user can see a simulated environment, usually displayed on a stereoscopic display. Headset movement is tracked and the resulting image is modified depending on the position of the user relative to tracking devices.

Today's most common virtual reality devices are:

- Oculus Rift,
- HTC Vive.

Both headsets offer quite similar experience in the virtual environment.

A. *Oculus Rift*

Oculus Rift is considered to be the first major virtual reality headset that showed the potential of VR with using modern hardware. It is developed and manufactured by Oculus VR.

Their headset Oculus Rift DK1 was one of the first headsets available for virtual reality development. This version featured a resolution of 1280x800px, which is quite low compared to today's headsets. When wearing the headset some users had complaints about motion sickness, mostly due to low resolution and display refresh rate [3].

When using the headset, a raster grid can be could be seen, meaning the individual pixels were noticeable, which broke the immersion inside the virtual environment.

Figure 2. Oculus Rift VR Headset [3]

With more advancement in the technology came Oculus Rift DK2 and finally Oculus Rift CV (Consumer Version) (Fig. 2). Comparison of technical features of these devices is shown in table 1.

TABLE I.
BASIC HEADSET COMPARISON

Model	Resolution	Refresh Rate
Oculus Rift DK1	1280 x 800	60Hz
Oculus Rift DK2	1920 x 1080	75Hz
Oculus Rift CV1	2160 x 1200	90Hz
HTC Vive	2160 x 1200	90Hz

Users report fewer problems with motion sickness when using headsets with higher refresh rates.

B. *HTC vive*

The headset HTC Vive (Fig. 3) was made in collaboration with PC games giant Valve and works with Valve's mammoth gaming ecosystem, Steam. HTC packs in 70 sensors to offer 360-degree head-tracking as well as a 90Hz refresh rate – that's the stat that's key to keeping down latency, which is the technical term for the effect that causes motion sickness [4].

Figure 3. Oculus Rift VR Headset [4]

The most common applications for virtual reality are video games. However, there are also available commercial applications focused on certain fields, for example, medicine. These are however expensive or difficult to access. Our goal was to examine the options of modern headsets for virtual reality and develop certain testing applications, focused on education.

The HTC Vive comes with two motion controllers. These are held in both hands by the user, while inside the virtual environment. These controllers allow him to move around or interact with certain objects.

Virtual reality is intended for everyone, however, the minimal recommended age for using the headset is at least 13 years (for the Oculus Rift). What is puzzling is the fact, that most modern arcade rooms with virtual reality allow the minimal age of 8 years or even 5 years. At this age, children are still developing their eyesight and it is not recommended for them to use the virtual reality headsets.

We used the HTC Vive device while developing our example applications, in addition to a powerful computer using Intel Core i7 processor, 16GB RAM and NVIDIA GeForce GTX1070 graphics card.

C. *Play Area for Virtual Reality*

There are basically two options to set up your play area for virtual reality.

One way requires the user to have plenty of space to move around. The HTC Vive recommends an area of approximately 3.5m x 3.5m. This is due to the maximum distance between tracking base stations being 5m. The user can use this area to move around, for example, to reach for an object inside the virtual environment (Fig. 4). When the user wants to move outside of this boundary, the hand controllers mostly offer to teleport inside the virtual environment. A virtual grid is displayed to the user inside the headset when he is getting close to a boundary in the play area and thus letting him know his approximate position within the virtual environment.

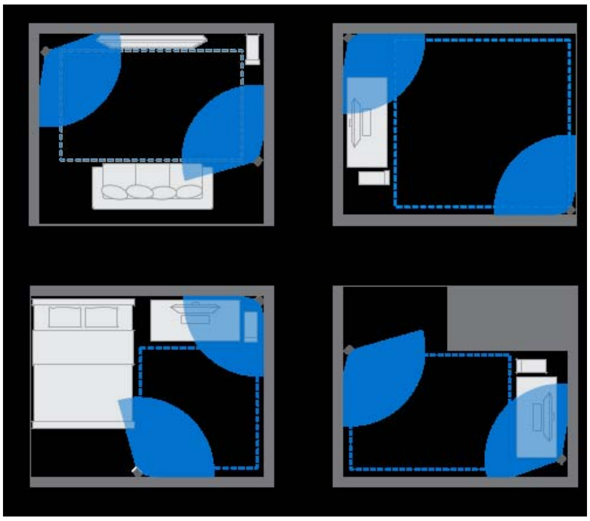

Figure 4. Play area arrangement [4]

The second option for the user is to sit in the chair while wearing the headset. This is most common approach when there is not enough space for the user to maneuver properly. Input devices can be mouse and keyboard or a gamepad. This is how the older Oculus Rift headsets worked (for example Oculus rift DK2). The new Oculus Rift Consumer Version has already the Oculus Touch controllers available.

Both variants are commonly used, meaning that our applications should support both.

However, in our case, we focused mainly on the first option, that gives the user more freedom of movement and allows better immersion in the virtual environment. More information about these options and setup can be found here. [2].

III. Application Development

Application for virtual reality can be developed using various tools. If the goal is to achieve a realistic look, one appropriate option is to use a modern game engine. A game engine is a tool used to develop modern games 3D and 2D applications.

Most common modern engines are:

- Unity 3D [5]
- Unreal Engine 4 [6]

Both engines have their pros and cons, although for our purposes we chose Unreal Engine. Mainly because of its support for the C++ programming language and the ease of achieving a realistically looking result. An example of the development environment in Unreal Engine 4 is shown in figure 5.

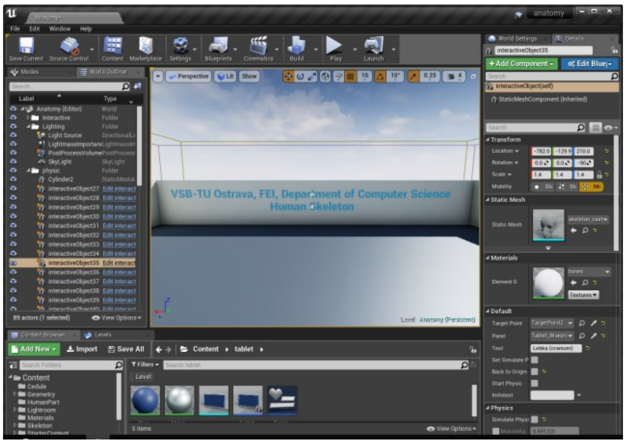

Figure 5. Unreal Engine 4 Editor

IV. Test Applications

For testing and demonstration purposes, several example applications were developed focused on different disciplines. Afterwards, these applications were tested on users. The topics were chosen to be challenging to implement and visualize, testing the capabilities of chosen tools and skills of developers.

A. Visualization of the human skeleton

The very first application was the visualization of the human skeleton. Our goal was to enable interactive inspection of individual bones that form the human body. For these purposes, a low-resolution model of the human skeleton was created.

To modify the skeleton model, we used an open-source 3D tool called Blender [7]. Individual bones were edited to suit our purposes and the final model was approximated to use roughly 500,000 triangles to form the surface of the skeleton (Fig. 6).

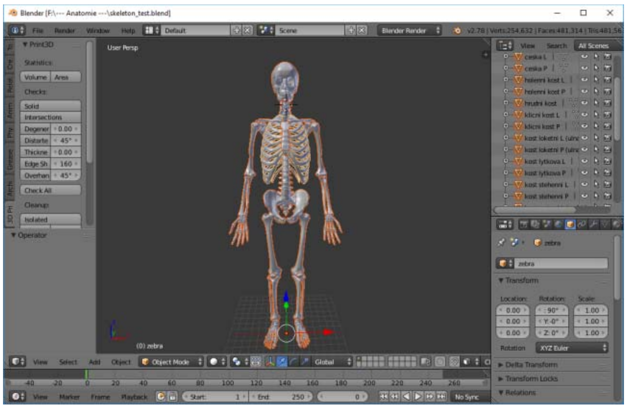

Figure 6. Human Skeleton Modification in Blender

Afterwards, individual bones were exported from Blender [8] using the FBX file format and imported into the Unreal Engine 4 [9]. Inside the Unreal editor, appropriate materials and collision models were applied to the models and placed into the virtual scene.

Inside the virtual reality, the user can inspect the human skeleton. Using interactive HTC Vive controllers, the user can interact with the skeleton [10]. For example, he can pick up individual bones, bring those closer to the user's head and inspect them up close, see Figure 7.

Figure 7. Practical testing with HTC Vive

For basic information about individual bones, a virtual tablet was created and placed into the scene. While holding a certain bone inside the virtual environment the user can pick up this tablet and read there some information about that bone.

For the purposes of testing, two variations of the skeleton were created. The first variation had the bones which always returned to their place after being let go, while the second skeleton had their bones stay in midair, allowing the user to disassemble the skeleton completely.

While testing this application, we encountered interesting moments. When the user found out that the second skeleton could be disassembled, the majority of the testing group immediately started creating their own version of the human skeleton.

In the next part, we tested objects that were physically simulated, meaning they could be thrown around. When the user lets go of these, physics simulation was activated and these objects behaved as if real.

Figure 8. Virtual table with human parts

These objects were placed on a table (Fig. 8) inside the scene. Turns out that this was not the best solution. While the users wanted to pick these objects up, they found that they could not reach so far, due to space limitation of the play area. In addition to that, some users wanted to lean on the table (visible only to them inside the headset), while reaching for those objects, which could potentially result in accidents.

B. Solar System Planets in VR

Another exemplary application was the visualization of planets of our solar system. The main goal was to display the individual planets in their relative sizes to each other so that the user could see the size difference between them. Thanks to the use of virtual reality user could also grab and closely inspect every planet's surface.

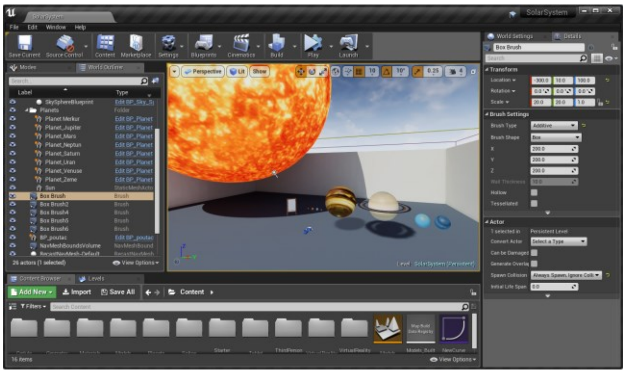

Figure 9. Solar System Planets

Due to the selected scale of the planets, of course, the sun had to be modeled in the same scale. So, in the end, the sun occupied a very large portion of the scene. Interestingly, when looking through the monitor, the size of the sun did not surprise anyone. When using the headset, however, most users were amazed by the size of the sun compared to other planets (Fig. 9).

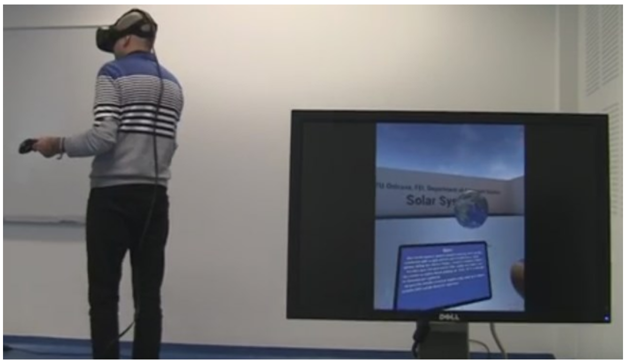

Figure 10. Solar System Planets

In the scene, there was also a virtual tablet (Fig. 10), which displayed the basic information about each planet. So, the user could grab any planet, view it, and read the basic information about it.

C. Interactive architectural visualization

Another experimental application is in the field of architecture. Our effort was to visualize the architectonic project as realistically as possible, and thus test the possibilities of realistic rendering in conjunction with virtual reality.

At the beginning, we chose the appropriate project [11], which in our case was the project of a family house in ArchiCAD (Fig. 11).

Figure 11. Family house - 3D model (ArchiCAD) [11]

First, it was necessary to remodel the entire scene (Fig 12) in a mesh-based modeling software, and then transfer the individual models to Unreal Engine (Fig. 13). Because of the number of individual objects and the complexity of the whole scene, optimization was a big part of this project.

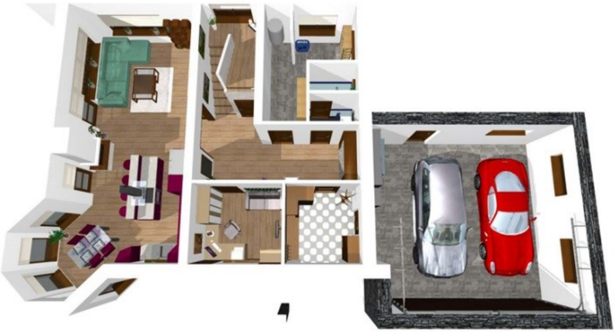

Figure 12. Family house - 3D model (ArchiCAD) [11]

During the rendering of more demanding scenes, the refresh rate of the screen may drop from HTC Vive native 90Hz to 45Hz or less. With this drop in the rendering frequency, there is already visible tearing, which causes unpleasant feelings and motion sickness to some users (kinetosis) [12]. To avoid such problems, optimizations are inevitable, such as reducing rendering complexity and level of detail [13], [14].

Figure 13. Visualization of the family house in VR (Unreal Engine 4)

Thanks to the use of virtual reality, the user can move freely both inside and outside the house and get a better idea about sizes of individual rooms, furniture, and overall layout. The realistic look further enhances the entire experience, when walking through the building the user can feel like he is in a real house (Fig. 14).

Figure 14. Interior of a family house (Unreal Engine 4)

Compared to the original project, which included only a few renderings and floor plans of the house, using virtual reality to move anywhere in the house is much more realistic and prominent.

For instance, when somebody wants to build a house and must make decisions based on a ground plan and several rendered images, he can never form a real picture of the finished house.

However, if he could walk through the planned house in virtual reality, he could create a very realistic image of what the final house will look like. He could also have remarks about individual parts before construction, which will increase the customer satisfaction.

D. The Dukovany Nuclear Power Plant

The last and the biggest project is the visualization of the Dukovany nuclear power plant [15]. Nuclear power plants are among the structures that are problematic to view. Certain parts of the nuclear power plant cannot be publicly inspected either because of security or because of health reasons. An example of such place is a nuclear reactor, as only trained employees have access there.

We used panorama imagery as a reference for modeling every individual part to gain some insight into the layout of buildings and interiors (Fig. 15 and Fig. 16).

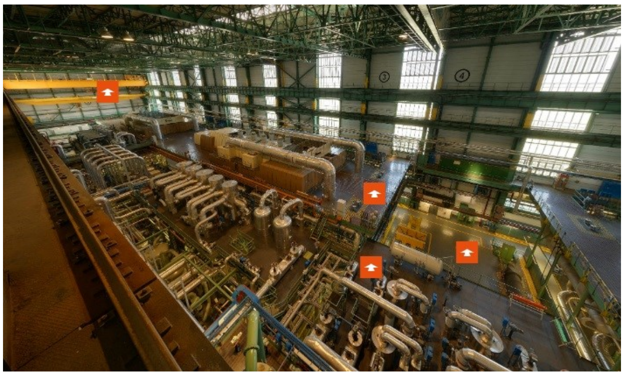

Figure 15. 360-degree panorama image of machinery room (Unreal Engine 4) [15]

Modeling according to such reference is more complicated and sometimes it is not easy to guess the right proportions and positions. Because of the complexity of the whole project we had to study basic principles and modeling only the most important parts. There was also a problem with looking up appropriate documentation and description for individual parts of the nuclear power plant, which complicated the creation process.

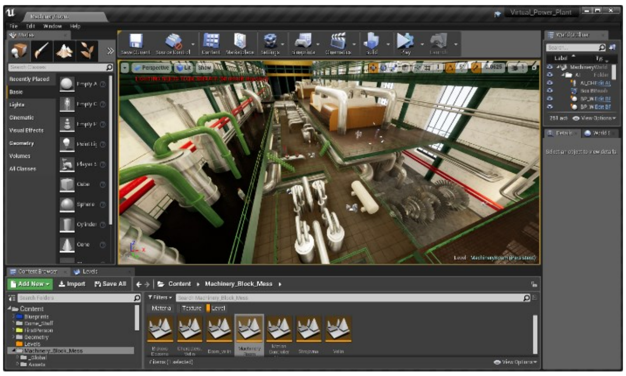

Figure 16. 3D visualization of the machinery room (Unreal Engine 4)

Similar problems arise when modeling the outdoor environment and the basic buildings, as the existing resources are relatively limited. Figure 17 shows a view from one of the cooling towers.

Figure 17. 3D visualization of the exterior of the Dukovany power plant (Unreal Engine 4)

V. CONCLUSION

All conducted experiments have shown that the use of virtual reality (HTC Vive) offers great opportunities for all scientific disciplines. It also offers big opportunities in the field of education, where it allows to visualize real and fictional scenes and thus allows viewing inaccessible places.

During testing, the described applications were tested on more than 1,500 people, and the results were always very positive.

ACKNOWLEDGMENT

This work was partially supported by the SGS in VŠB Technical University of Ostrava, Czech Republic, under the grant SP2017/36, and grant IRP-RPP No. RPP2017/200.

REFERENCES

[1] Ch. Crawford, "SCRAM: Internet Archive", Retrieved August 20, 2017, from https://archive.org/details/a8b_SCRAM_1980.

[2] T. Parisi, "Learning Virtual Reality, Developing Immersive Experiences and Applications for Desktop, Web and Mobile", *O'Reilly Media, Inc,* 2015, ISBN 063-6-920-03846-7.

[3] Oculus Rift, Retrieved August 20, 2017, from https://www.oculus.com/rift/.

[4] HTC Vive, "Discover Virtual Reality", Retrieved August 20, 2017, from https://www.vive.com/.

[5] Unity - Game Engine. Retrieved August 15, 2017, from https://unity3d.com.

[6] Unreal Engine. Retrieved August 15, 2017, from https://www.unrealengine.com.

[7] Blender - Home of the Blender project. Retrieved August 15, 2017, from https://www.blender.org.

[8] J. M. Blain, "The Complete Guide to Blender Graphics: Computer Modeling & Animation, Third Edition", *CRC Press*, 2016, ISBN 978-1498746458

[9] J. Plowman, "3D Game Design with Unreal Engine 4 and Blender ", *Pack Publishing*, 2016, ISBN 978-1-78588-146-6.

[10] M. McCaffrey, "Unreal Engine VR Cookbook: Developing Virtual Reality with UE4 (Game Design)", *Pearson Education, Inc*, 2017, ISBN 978-0-13-464917-7.

[11] M. Somerlikova, "Projekt rodinného domu v programu ArchiCAD", SPŠ Stavební Ostrava, 2017.

[12] MedicineNet, What Causes Motion Sickness, Retrieved August 15, 2017, from http://www.medicinenet.com/motion_sickness.

[13] A. Cookson, "Unreal Engine 4 Game Development in 24 Hours", *Pearson Education, Inc*, 2016, ISBN 978-0-672-33762-8

[14] J. Lee, "Learning Unreal Engine Game Development",*Pack Publishing*, 2016, ISBN 978-1-78439-815-6

[15] CEZ, "Nuclear Power Plant Dukovany", Retrieved August 15, 2017, from https://www.cez.cz/.