Virtual Reality System as a Tool for Education

V. Kovalčík, J. Chmelík, M. Bezděka and J. Sochor
Masaryk University
Botanická 68a
602 00, Brno, Czech Republic
vit.kovalcik@mail.muni.cz; jchmelik@mail.muni.cz; mbezdeka@mail.muni.cz; sochor@mail.muni.cz

ABSTRACT

Education in the field of computer graphics is attractive, but the necessity of strong theoretical background and the amount of work needed before achieving a non-trivial output could be discouraging for many students. To cope with inherent complexity of computer graphics applications, we designed a system intended to reduce implementation burden and help the students to achieve interesting results in less time. In our labs we use an in-house virtual reality system, which has been gradually improved by the work of our students and staff for a period of more than eight years. The foundation of the system is a modular architecture that allows rapid assembly of prototyped applications as well as easy implementation of extensions. Computer graphics students with even a fundamental level of knowledge are encouraged to participate in the lab projects. Working on system that allows rapid prototyping and shortens the path to achieving interesting visible results, builds up the students' motivation and it is a key element of a continuous improvement during the study.

Keywords:
Virtual Reality, Computer Graphics, Education.

1 MOTIVATION

Computer graphics is regarded as one of the most attractive parts of computer science. However, at the beginning students often do not realize the amount of work that is necessary to create a new non-trivial graphics application. When they realize this fact, it might decrease their motivation and diminish the enjoyable part of the learning. However, much of the work has been already done in the past. For their lab assignments, students can use existing graphical libraries, which implement the basic and repetitive parts of the applications and leave the student to do the creative part of work. Unfortunately, the higher level interface is often missing. which would allow the user to connect the modules into a loosely coupled systems and to support assembly of complex applications from partial solutions.

To achieve better cooperation and allow rapid prototyping of virtual reality applications we designed a modular system VRECKO. Students can extend this system and build upon the work of their colleagues from previous years. This is similar to the educational concept of fixing and extending application used by Costantini et al. [CMC09]. The products of students' work are usually rich visual applications. This is also motivating – it corresponds to the findings of Guzdial et al. [GS02], stating that creating multimedia application instead of working on abstract algorithms builds up the motivation.

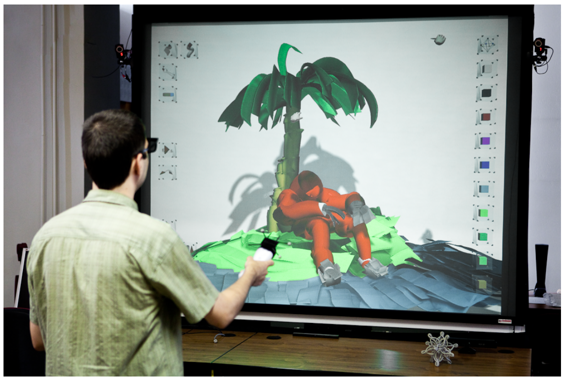

Figure 1: 3D painting in a virtual environment controlled with Wiimotes™, with position of user's hands being tracked by the OptiTrack™system. The application was developed using the VRECKO system.

Similar long-term projects were created in previous years. ShadowLight [Lee05] is a modular system for rapid prototyping in virtual reality created at University of Illinois. Pettersen et al. described a Virtual Laboratory [PJS03], a 3D network-distributed virtual environment for education of computer graphics programming. Similarly, a system VRUT (developed at the

Czech Technical University in Prague) was designed as an experimental VR system and it is used for education as well as in industry. For the augmented reality applications, the STUDIERSTUBE [FLSG98] software framework gained much attraction in the past decade.

2 HISTORY AND CONCEPT OF THE SYSTEM

Virtual reality system VRECKO[1] , was designed to support the research in Human-Computer Interaction laboratory at Masaryk University. The system was originally developed as a part of two dissertation theses. The first one modeled the concept of "object with changeable roles" [Mar04] and second one used VRECKO framework for development of applications for prototyping various interaction tasks in virtual reality [Fla05].

From educational point of view, common graphic applications are usually designed for a specified purpose and difficult to extend beyond the original scope, while graphic engines and graphic libraries are too generic for rapid implementation. We tried to strike the balance and develop a system that fulfills the basic tasks in interactive applications and offers a high degree of flexibility.

The system allows the programmer to create scenes composed of objects and to define the behavior of each object by adding one or more roles, called *Abilities*. The potential of *Abilities* range from changing object color or adding a grass onto a surface to the navigation and movement of a car in a virtual city.

The user can designate connections between the *Abilities* which are then able to communicate using messages, originating usually from some user's action. Alternatively, an *Ability* can be scheduled to execute in periodic time intervals without the need of any external message.

Recently, the system is used as a basis for several research projects and in many undergraduate projects.

3 USAGE ADVANTAGES

The usage of our system is advantageous in many ways. First of all, it supports various kinds of input/output devices. The output devices include stereo projection on a large screen or head-mounted displays. The input devices include PHANTOM® haptic devices, OptiTrack™camera tracking system, Nintendo® Wii™or Microsoft® Kinect™among many others. A support for new devices can be quickly added when necessary.

[1] http://decibel.fi.muni.cz/wiki/index.php/VRECKO

All devices are regarded as special objects that are able to communicate via messages. Using this approach, the same application can be controlled by the OptiTrack™system and displayed on the stereo projection wall, as well as with a mouse and on a common computer screen. The concept of replaceable device objects allows students to work either in the laboratory or on their computers at home and also to present the final work outside the laboratory.

Another advantage is standardization of the environment which is easily understandable and the knowledge is transferable. If a student decides to do their work in the laboratory room, they can discuss the issues with their colleagues, which strengthens the spirit of the teamwork and speeds up the process of passing experience and learning new things.

4 LEARNING TOPICS

At the beginning of a semester, the students are offered with a list of possible research topics. Alternatively they can also come up with their own suggestions. Some projects may span over more than one semester, and may even converge to bachelor or master theses.

Throughout the years several main branches of the system utilization had evolved, with the needs for specific extensions. In the following parts we will describe the main directions of the development and give examples of student's contributions.

4.1 Editor

The oldest branch in development is the Editor application for creating and editing 3D scenes [KFS07]. The Editor itself uses a set of specifically designed tools implemented as *Abilities*. Students can easily implement their own tools extending the editor capabilities.

The common tools for interaction in VE are already implemented as a core functionality of the Editor. Such tools include: addition of objects to the scene; object movement; path definition for movement and animation of individual objects; camera setup and movement; removal of objects.

In the previous years, a number of high quality specialized tools have been developed by the students as a practical part of their bachelor and diploma theses. The most notable examples are:

- *Room Edit* for editing 3D room schemes, i.e. editing walls, creating and moving windows and doors and coloring the objects.
- *Constrained Movement* for fast placement of objects with pre-edited snapping capabilities.
- *FFD Editor* for free-form deformations of objects (see Figure 2).
- *Connect Editor* for seamless automatic aligning and connection of objects geometries.

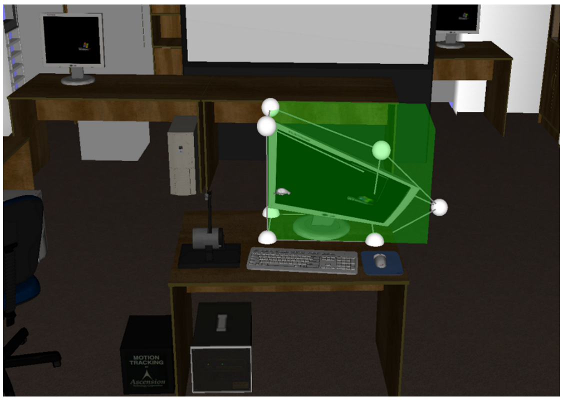

Figure 2: The Editor applied to the model of the HCI laboratory. The virtual LCD monitor can be deformed using a free-form deformation tool.

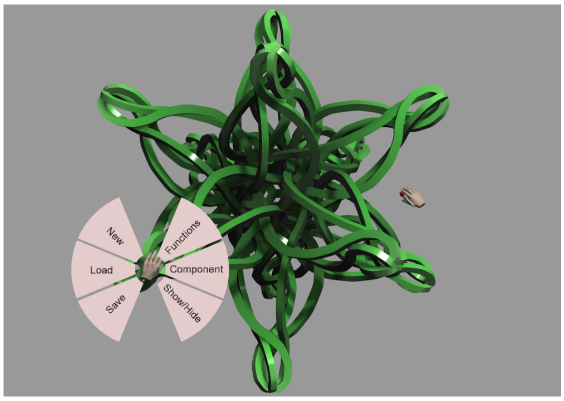

Figure 3: Designing a mathematical sculpture based on a dodecahedron.

Students participating in this branch of development learn to implement and improve low-level algorithms for manipulation with objects, such as positioning, shaping, lighting and viewing, and another similar operations typical for 3D editors. By solving the elementary tasks students get acquainted with practical usage of quaternions, matrix transformations and other mathematics background frequently used in computer graphics.

4.2 Artistic applications

Another optional track for students' work is development of extensions for artistic applications (see Figure 1). Similarly to the *Editor* branch, there is also a layer providing common functionality. Therefore, students do not have to re-implement the code common to many artistic applications like basic GUI, objects manipulation, color selection, etc.

Currently implemented modules include:

- brush strokes based 3D painting,
- dance visualization,
- design of mathematical sculptures (see Figure 3),
- abstract performace art based on Perlin noise,
- exploration of chaotic attractors and fractals.

By implementing new art techniques, students can learn various topics of computer graphics. Availability of various input devices (like OptiTrack™, 3D mouse or Wiimotes™) also forces students to think about user interfaces and usability aspects in virtual environment.

4.3 Artificial World

Third direction of development is a set of applications for creating and displaying artificial worlds (see Figure 4).

Technically, the virtual world is based on a set of XML files, which can be arranged in a tree. Each of the files contain one level of a world and references to the files containing the details.

The XML files hierarchy is created by a system of independent generators. Because of the independence of the generators, students have freedom to choose their preferred programming language and programming environment.

The currently implemented generators are:

- Planet generator allowing the user to generate the whole planet's surface by approximating the natural laws. Written in C.
- City generator, able to create a layout of a city on a given height-map and decide where to put roads and parcels and determine their usage. Written in Python.
- Building generator that can take parcels as the input and generate appropriate buildings or building blocks. Written in Delphi.

The freedom to choose the programming language has demonstrated the diversity of students' preferences. From a maintainer's perspective, this is not a perfect solution, but the generators are written in much less time, which is beneficial for the educational aspects, and each of the generators can always be easily replaced by a different one.

The generated world can be visualized in the VRECKO engine, but also enhanced by a number of plug-ins. Currently, we are using plug-ins for volumetric clouds, flocks of birds and car traffic simulations, that bring the city to life. We have also a parametric stochastic L-system able to generate high quality trees. All of the mentioned plug-ins were created by students. In this track the students learn mainly the processing of huge amounts of data stored in complex data structures.

4.4 Haptic interaction

Another set of applications use the VRECKO engine to perform fast multi-contact collision detection and

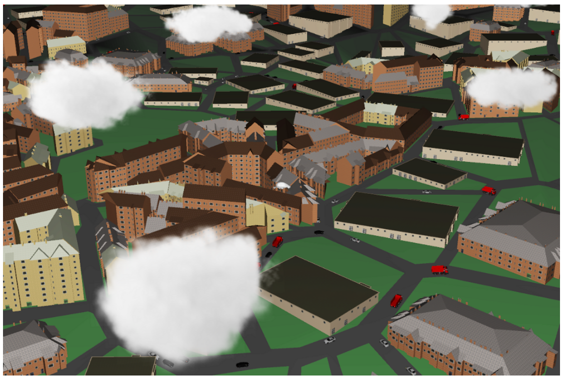

Figure 4: The Artificial World application, incorporating long-term work of many students.

to simulate haptic feedback, see [KKS11]. Projects cover the issues of different data preprocessing and different data structures for collision tasks evaluated in the scene. The collision detection algorithms are designed to be fast enough for the use with the SensAble® PHANTOM®.

While developing haptic algorithms, students learn how to create effective and optimized code which does not depend on the high performance of latest computer hardware and particularly the graphics cards.

5 EXPERIENCES AND RESULTS

The virtual reality and computer graphics in general is an attractive field for students. The chance to create interesting output alongside with the possibility to work with rare equipment presents a motivation to do better work. Aside from semester-long lab assignments, there were already more than 25 successfully defended bachelor and master theses that were based on the VRECKO engine.

The source codes of the best modules are added into the main distribution of the VRECKO engine and newcomer students can use them and learn from their code. The best applications are also presented to the public during numerous excursions, among others to high-school students interested in the study of Informatics. The fact of their work being actually used also manifest as a motivation boost for participating students.

The results proved to be attractive and the demonstrations of student's works draws attention of many prospective students to our laboratory and to our school. We do not collect any data confirming the shift in interest, but we already encountered students for which the excursion was a strong impulse to enroll the computer graphics program or other study programs at our school.

6 CONCLUSION AND FUTURE WORK

We have presented a virtual reality system, which is used for research and for education of students of the computer graphics program. Its modular architecture and number of main directions of development allows students with almost any programming skill to participate in the development either by writing a module or assembling an application based on existing modules.

The core of the system proved to be well designed and it is essentially the same as it was in its very beginning, more than eight years ago. It is stable and the most of the older modules are still compatible with current versions and can be utilized by present students. In our opinion, the effort put into building an universal VR engine has paid off and now we can use it for research and education as well as for excursions to attract new students.

Nowadays, we are continuously extending the system and we plan to prepare a set of generic task assignments to cover the educational topics more evenly and for various levels of education. During design and evaluation of new art modules, we are also collaborating with students of fine arts.

7 ACKNOWLEDGEMENT

This work has been supported by the Student Project Grant at MU under the research program MUNI/A/0922/2009

8 REFERENCES

[CMC09] G. Costantini, G. Maggiore, and A. Cortesi. Learning by fixing and extending games. In *Proceedings of Eurographics 2009*, 2009.

[Fla05] Jan Flasar. *Interaction Techniques in Non-Immersive Virtual Environments*. PhD thesis, Faculty of Informatics, Masaryk University, Brno, 2005.

[FLSG98] A. Fuhrmann, H. Loffelmann, D. Schmalstieg, and M. Gervautz. Collaborative visualization in augmented reality. *Computer Graphics and Applications, IEEE*, 18(4):54 –59, jul/aug 1998.

[GS02] Mark Guzdial and Elliot Soloway. Teaching the nintendo generation to program. *Commun. ACM*, 45:17–21, April 2002.

[KFS07] Vít Kovalčík, Jan Flasar, and Jiří Sochor. Extensible approach to the virtual worlds editing. In *Computer graphics, virtual reality, visualisation and interaction in Africa / AFRIGRAPH'07*, pages 31–37, Grahamstown, South Africa, 2007. ACM.

[KKS11] Vít Kovalčík, Lukáš Kalčok, and Jiří Sochor. Modular system for haptic interaction with multi-contact collision detection. In *Proceedings of the 11th International Conference on Computational Science and Its Applications (ICCSA)*, pages 24–32, 2011.

[Lee05] Kalev H Leetaru. Shadowlight: an immersive environment for rapid prototyping and design. In *Proceedings of SPIE*, volume 5664, pages 606–615. Spie, 2005.

[Mar04] Lubomír Markovič. *Object with Roles*. PhD thesis, Faculty of Informatics, Masaryk University, Brno, 2004.

[PJS03] L. W. Petterson, N. Jensen, and S. Seipel. A Virtual Laboratory for Computer Graphics Education. In *EUROGRAPHICS 2003 Education Presentation*, 2003.