

VRAM

a Virtual Reality Aided Modeller

Holger Regenbrecht, Ernst Kruijff, Dirk Donath, Hartmut Seichter, Jakob Beetz

Abstract

This article describes VRAM, short for Virtual Reality Aided Modeler. VRAM is a conceptual design tool supported by Virtual Reality technology and an ongoing testbed for theory and methodology in the field of three dimensional user interfaces (3DUIs). The outcomes from the project should consist of an intuitive and comprehensive immersive surface modeler, next to a set of taxonomies and guidelines for the development of 3DUIs. Based on a modular structure, VRAMs program architecture allows the easy extension of functionality. The application consists of the seamless integration of four main functionality modules, namely system control, viewing & browsing, editing and modeling. Based on the premise of portability, the software environment runs on both SGI Irix and MS Windows NT platforms. To be relatively independent in developing 3DUI techniques, the VRAM environment supports a multiplicity of input and output devices. Due to the focus on immersive modeling, tracking devices, head mounted displays and stereoprojection devices are the main I/O channels.

In this article, we will primary focus on the functionality of the virtual modeller.

Introduction

The strictly related fields of modeling supported by Virtual Reality (VR) technology and three dimensional user interaction in Virtual Environments (VE) has been subject to plenty of interest [4][1]. The apprehension of modeling in a VE is of great interest for designers, since it could allow the seamless fusion between design and review in a design process.

The VRAM software environment should run on multiple systems and should be available as open source for other interested developers. Even more important, the software should also provide the means to be practically used by our user group, which includes architects, designers and urban planners and should therefore run on a basic system. To fulfill this supposition, we implemented the software based on SGI OpenGL, Cosmo3D and Optimizer. Due to its software foundations, VRAM runs with the non-proprietary file format VRML97, and therefore also allows uniform file exchange between CAD software used by our user group and the VRAM environment. Our primary used hardware installations are desktop systems, a stereoprojection-wall and a head mounted display

system which allows a wide walking area due to our platform construction [5].

Functionality

At the current state of the software, 38 possible actions can be performed. They are grouped into four functionality modules, namely system controls, viewing & browsing, editing, and modeling. The geometric appearance and the arrangement of the functions are externalized and will be loaded online using the VRML97 format. With VRML97, the designer can use the same modeling tools for the virtual world (the digital model) and the interface. The function elements (actions) of the modules can be combined to realize the desired final application without any new compilation of the program itself. We will now give a short description of the respective modules.

The functions of the system control module are related to the issuing of a command in order to change the state of the system, or the mode of action. Examples are changing the field of view or other user specific parameters or exit, save, and load of virtual worlds or interfaces. People can easily create an interface in an external modeling program, because of the VRML97 basis. Interface designers model their ideas using the modeling program of their choice, export the results via VRML97, attach the appropriate names of the actions in the VRAM software environment to the specific interface elements (widgets) and use the interface in VRAM. Widgets are placed at either body-relative, platform-centered or world-centered places. In order to allow file-control, a basic VRML-like link structure has been implemented to build menu like interface structures by simply link the menu elements to each other. Several modes are available for adding and deleting elements, reloading them, and changing the location of the interface or world.

The viewer module in its essence resembles the

traditional walkthrough tool. It provides several means to support the user in navigating through the environment. This is achieved by either allowing the user to physically round on the platform construction or by using any of the implemented (artificial) traveling techniques, like point-and-fly, and seeking. In addition, several navigation (wayfinding) aids like a map and a compass have been made available.

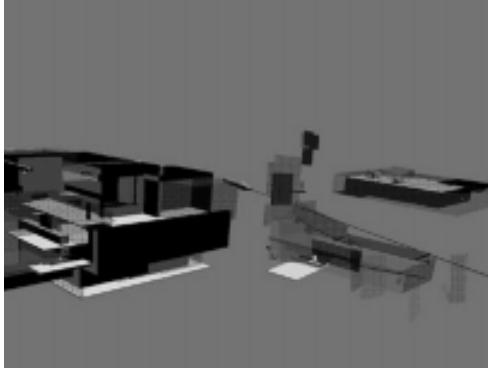
The editing module includes functions which are meant to adapt models which have been created in VRAM or imported from other applications, to rapidly create new (conceptual) designs. Due to the characteristics of conceptual design, the user requires a tool which is fast and relatively easy to handle. The user has to control the parameters of operations in an approximate, direct way. The editor allows the translation, rotation and scaling of objects, constrained in one up to three choosable dimensions. These constraints are necessary for more accurate operations, like moving an object straight upwards (constrained to up-direction). Furthermore, basic grouping, cut, copy, and paste functions are also supported.

Based on previous experiments at our university with approximate modeling using voxels in voxDesign [3][5], we developed a very simple modeling function called warting, which provides



Image 1:
photo of a user working on
an architectural mode
virtual environment shown
from the physical cameras'
perspective, user view
small image at upper left

Image 2:
user view during a
modelling session



a quick way to draw with a chosen VRML97 object onto existing surfaces. The user is, however, also able to freely sketch into space by placing objects are the current location of the input device. Additionally, a first approach to surface modeling and a (by Deering's toothpaste modeling from the HoloSketch application inspired [2]) method of creating spline-extrusion tubes directly into space.

Conclusion and discussion

VRAM currently offers a basic level of functionality. To make it a fully usable conceptual design tool, several logical extensions can be envisioned. The task domain requires a broader range of functions which has already initiated the development of new tools and supporting interaction techniques. This approach also includes the exploration and imple-

mentation of new devices. Among the new developments, we are working on two-handed interaction with gloves, refinement of the surface-modeller and additional ways of creating traditional and new interactive architectural forms and spaces.

The ongoing use of VRAM in educational situations will be extended as it gives us continuous feedback. By increasing the number of installations in other architectural facilities we also hope to be able to involve additional contributors to the applications' code itself.

References

- 1 Bowman, D., Kruijff, E., LaViola, Poupyrev, I., **The Art and Science of 3D Interaction**, tutorial notes, IEEE VR 2000, New Brunswick, 2000.
- 2 Deering, M., **The HoloSketch VR Sketching System**. Communications of the ACM, 39(5), 1996.
- 3 Donath, D., Regenbrecht, H. **VRAD - Virtual Reality Aided Design in the early phases of the architectural design process**, proceedings of caad futures '95, Singapore, 1995.
- 4 Hand, C., **A Survey of 3D Interaction Techniques**. Computer Graphics Forum, 16(5), 1997.
- 5 Regenbrecht, H., Donath, D., **Architectural Education and Virtual Reality Aided Design (VRAD)**. Designing Digital Space. D. Bertol (Ed), John Wiley and Sons, New York, 1997.

Regenbrecht, H., Kruijff, E., Donath, D., Seichter, H., Beetz, J.
InfAR - computer science in architecture
Bauhaus-Universität Weimar
caad@archit.uni-weimar.de