Scientific Visualization Department
Supercomputer Center (HLRZ)

RoboVis - a Scenario for Using Virtual Reality Techniques in Learning Robot Development

Berthold Kirsch
Ipke Wachsmuth *

Scientific Visualization Department of HLRZ
German National Research Center
for Information Technology
Sankt Augustin, Germany

* On leave from University of Bielefeld,
Faculty of Technology, Bielefeld

Uwe Schnepf

AI Research Division
German National Research Center
for Information Technology
Sankt Augustin, Germany

VISWIZ Report
4/1993

To appear in: Proceedings of the Int. Workshop on Graphics & Robotics, Springer, 1993

RoboVis – a Scenario for Using Virtual Reality Techniques in Learning Robot Development

Berthold Kirsch †, Uwe Schnepf ‡, Ipke Wachsmuth*†
† Scientific Visualization Dept. of HLRZ
‡ AI Research Division
GMD
53757 Sankt Augustin
Germany

Abstract. The context of this paper is the design of an appropriate working environment for the development of learning robots. We are interested in robots which are able to structure incoming sensor data on the basis of internal reference schemes and robot tasks. For efficiency reasons, we want to perform the learning cycles in simulated environments explored by the robot. In this paper, we outline how advanced visualization and interaction techniques as developed in the field of Virtual Reality could be employed to study the development and properties of the internal data of a semi-autonomous robot, as well as the learning process itself.

1 The Scenario

Robots of the forthcoming generation are envisioned to have greater autonomy in getting organized in their environment, and they will be able to make task-dependent decisions on the basis of available world data. These robots will be learning systems which gain information about their environments individually, and will develop and maintain their own world models of the environment on the basis of their sensorimotor capabilities. This way, they create their own "Merkwelten", i.e. simple internal representations of corresponding entities in the real world.

In our system, various processes are used to set up a group of communicating agents which cover different aspects of the robot-world-user interaction, e.g. simulation, visualization, interaction, and control. In our current scenario, a physical robot serves as a template for a simulated robot model: a virtual robot in a virtual environment. A robot controller is able to control either the physical or the virtual robot without any further modification. The virtual

---

*On leave from the University of Bielefeld, Faculty of Technology, 33501 Bielefeld.

environment is intended to show nearly the same characteristics as the natural environment of the physical robot.

The virtual environment can be any artificial room, and the virtual robot will navigate through this room. In this room, there are one or several users which participate in this scenario through the use of headphones and data gloves. The user is able to inspect the robot's internal representations and to switch between different robot views. Also, it is possible for the user to interact with the robot in their shared environment, either physical or simulated.

In the following sections, we will outline our approach, and present the technical solutions to the problems discussed as well as initial progress which has been made.

2 The Approach

Our goal is to support the developer of learning robots with advanced methods for controlling, inspecting and manipulating the learning processes of these robots. To this end, we want to enable the developer to actually experience the sensorimotor capabilities of the robot in terms of both raw data as well as learned data abstractions performed by the robot controller (the *robot data view*). This robot sensor and control data has to be translated into representations accessible by human senses. In addition, we want the user to be able to experience and manipulate internal states and processes of the robot controller (the *Merkwelt view*). Finally, we want to equip the user with advanced visualization and interaction techniques in order to provide natural ways of inspecting and testing the robot's behaviour (the *user view*).

Our overall system consists of different modules which communicate via Unix or Transputer process communication facilities such as sockets and transputer links. There are one or several robot control processes, one world simulation, and one or several visualization processes. Additional processes, which can be active at a given time, serve to drive interaction devices such as data glove, joystick, and tracking sensors. The various processes can be downloaded on a computer network consisting of Unix workstations and transputers (cf. Fig. 1). It is planned to incorporate a real robot as hardware platform to actually test and evaluate the simulation results.

3 The Working Environment

In this section we describe how we want to realize different aspects of the working environment mentioned above. As a basis, we use a physical robot as described below. First we have to model the robot, its physical sensorimotor capabilities and its environment to realize the simulation aspect of our work. Second, we need interface communication channels between the different modules such as robot controller, world simulation and visualization. Third, we

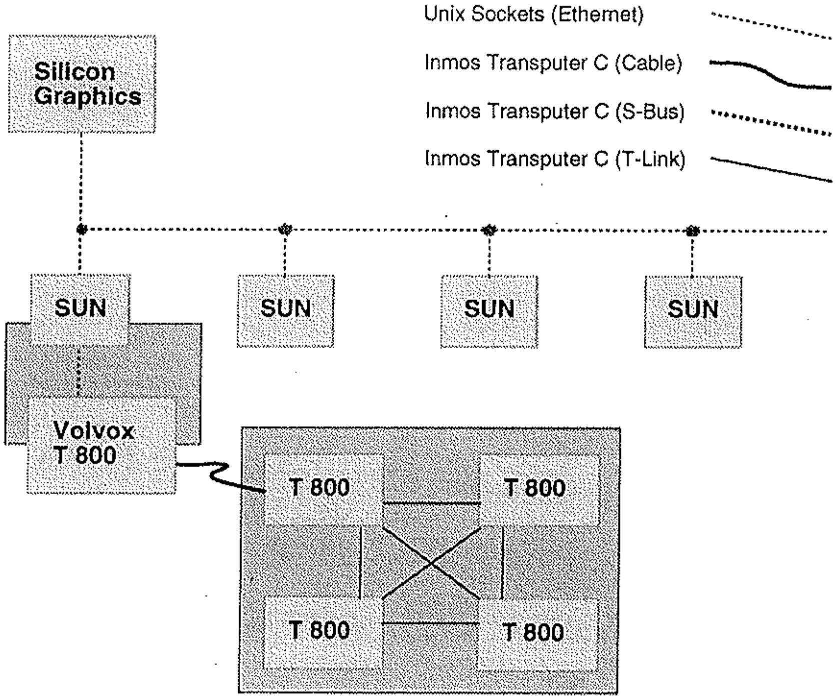

Figure 1: Our current hardware configuration for robot control processes, world simulation, and visualization.

describe three different views, the *robot data view*, the *Merkwelt view*, and the *user view*, which enable the user to obtain taylored visualizations of the robot and its activities.

3.1 The Physical Robot

The physical robot is equipped with four wheels driven by two separated stepping motors, and one or several of the following sensors: ultrasonic sensors, touch sensors, simple light intensity sensors, odometric sensors, laser range finder, and compass. We have started with using ultrasonic and touch sensors because of their simplicity and robustness as well as the low complexity of returned data.

3.2 The Virtual Robot

The virtual robot has similar sensorimotor characteristics as the physical robot plus additional abstract sensors which only exist and function in the virtual versions of the robot and its environment. These abstract sensors can be thought of as, e.g., a data glove sensor which is able to transmit various commands given by a human instructor.

3.3 Modelling

Modelling can be divided into two categories, (1) geometric and (2) physical modelling of the robot, its sensors and motors, and the robot's environment. For geometric modelling, we use standard techniques from computer graphics to create the data model to be visualized (cf. Foley et. al. [1]). This data model is the basic reference model for the visualization taking place when adopting the user view.

The physical model of the robot and its environment will be developed to realistically simulate sensorimotor behaviour of the physical robot as well as changes in the simulated environment due to interactions of the user with the working environment. For sensorimotor simulation, we apply specific techniques for the simulation of the physical effects of various robot components such as ultrasonic, light and touch sensors, as well as motors. We make use of raytracing-like techniques for the simulation of touch and ultrasonic sensing. For imposing changes on the scene, we use Virtual Reality interaction techniques, and other control devices such as joystick or mouse.

3.4 Communication

Process Interface: The main modules of the system are the physical robot, the robot controller, the virtual robot in its virtual environment, and the visualization. All these modules can be linked together through a parallel message passing-based programming system based on distributed and cooperating Unix and transputer processes (cf. Schnepf et. al. [2]). The Unix-based processes such as visualization and interaction are added to the overall system via Ethernet sockets, the transputer processes such as robot control and simulation via physical links (for inter-transputer communication) and "virtual" links (for outer-transputer communication). The robot controller communicates with the physical robot in the same way as with the virtual robot. For the robot's controlling behaviour there is no difference between communicating with the physical or the virtual robot. The learning process, which is part of the robot controller, is independent of which robot (virtual or physical) is actually used. The visualization module receives its data either from the robot controller only (when using the physical robot) or from the robot controller and the environment (when using the simulated robot). Then these data is processed and displayed depending on the selected view.

User Interface: Since the working environment is adaptable to different parameters (physical or virtual robot, physical or virtual sensors used) the interface to the user interacting with the robot and the environment must be configurable as well. This may be done via keyboard commands, mouse or user interface tools which consist of buttons, sliders, browsers, or input fields. Using Virtual Reality equipment such as the data glove allow a more straight-forward approach to interact with the robot and the environment. A head-mounted display allows to explore sensor data in a more direct way. For instance, a visual grasp of robot sensor data could support the system-user cooperation in solving a given problem, e.g., obstacle detection, or maneuvering.

3.5 Visualization

The robot and the environment are modelled as different sets of simple planar surfaces. The robot shape can be scaled, and an arbitrary number of sonar and touch sensors can be defined and placed onto the robot shape. The model can be visualized as a wireframe picture as well as a more elaborated picture including static, precalculated light intensities based on radiosity. The robot currently is visualized by a simple model of a real robot. Later on, this simple representation will be replaced by a more realistic model of the physical robot, or by texture mapping taking from a real robot. This is just to ensure that the user is operating in a more natural and familiar environment rather than in a uniform and abstract one.

The user-robot communication is the communication between learning robot and the human supervisor. The supervisor's knowledge about the world shall support the robot's learning process. For this purpose, the human needs to know about the internal robot control data, i.e. the 'Merkwelt', to see in which area there is a lack of knowledge. The robot then can be modified (e.g. replacing or adding sensors, or defining a new control structure) or brought into certain new situations to be dealt by the help of the supervisor. This process will be a new area of research.

We want to develop methods for inspecting internal properties and activities of the robot controller. For this, we consider two different views, the *robot data view* and the *Merkwelt view*.

The robot data view: The robot data view requires new visualization techniques to translate and display non-human sensory information into a form understandable to humans. For instance, it is necessary to display the dynamic characteristics of ultrasonic beams. The robot data view is used to visualize the data from the different (physical or simulated) sensors of the (physical or simulated) robot. For this purpose, we have to transfer the non-visual quality of the (physical or simulated) sensors into visual representations understandable to humans. This will be done by constructing spheres the shape of which will depend on current sensor readings. Additionally, some form of world knowledge will be used to construct data sets more abstract than provided by raw data only. Adopting the robot data view, the user can examine and interpret the data which form the basis of the robot's current activities.

The Merkwelt view: The Merkwelt view expresses the internal robot "knowledge" gained by ongoing learning processes. The idea to provide the Merkwelt view originates from the incorporation of machine learning techniques into the robot controller. The learning process and the knowledge state of the robot, as it is based on the robot's sensing and acting capabilities, is difficult to understand by the user and therefore will be translated into visual categories. How to represent the Merkwelt view will be a new research topic to be dealt with in cooperation between robotics and graphics. The main focus herein will be the visualization of the gradually growing knowledge of the robot controller about its (physical or simulated) environment formed on the basis of ultrasonic sensor readings. The dynamic parameter settings of the learning algorithm class employed (i.e. neural networks and reinforcement learning techniques) will be visualized and animated.

The user view: Apart from developing methods for inspecting internal properties and activities of the robot controller, we need a visual way to inspect the overall scene. When using the robot simulation, the user view is to provide similar external viewing and interaction capabilities as in a physical testing environment. In this view, the user underlies no restrictions with respect to physical size, position, and movements. Further simulated devices such as abstract sensors and manipulation abilities for robot control algorithms may enrich the physical interaction capabilities. Through the use of interaction devices, the user has the ability to switch between the different views, in order to obtain as much information available about the learning system as possible.

4 State of Work

Work is divided into several work packages some of which have been realized at the time of this writing, and others which still have to be implemented (based on our existing robot, its sensorimotor capabilities, and the message passing-based operating system as described before):

Geometric modelling: The geometric modelling activities have begun using the `wavefront` data file format. We have modelled the robot shape in 3-D. Various sensors (e.g. sonar sensors) have been modelled in order to mount them onto the robot shape (cf. Fig. 2). 3-D models of several workspaces such as maze, building, and factory hall have been developed (cf. Fig. 3) to serve as virtual environment. It is possible to load different geometric models of the robot and its environment, and to use these different models for the simulation. The geometric models serve as the basic data set to compute visualization and illumination, and to perform the simulation of physical data such as obstacle collision and sonar sensing.

We are able to directly feed in the geometric model of the robot and the environment into the visualization system and can make use of different views and perspectives. Currently, the robot is shown as a simple cube, and the environment has simple perpendicular walls with homogeneous surface and

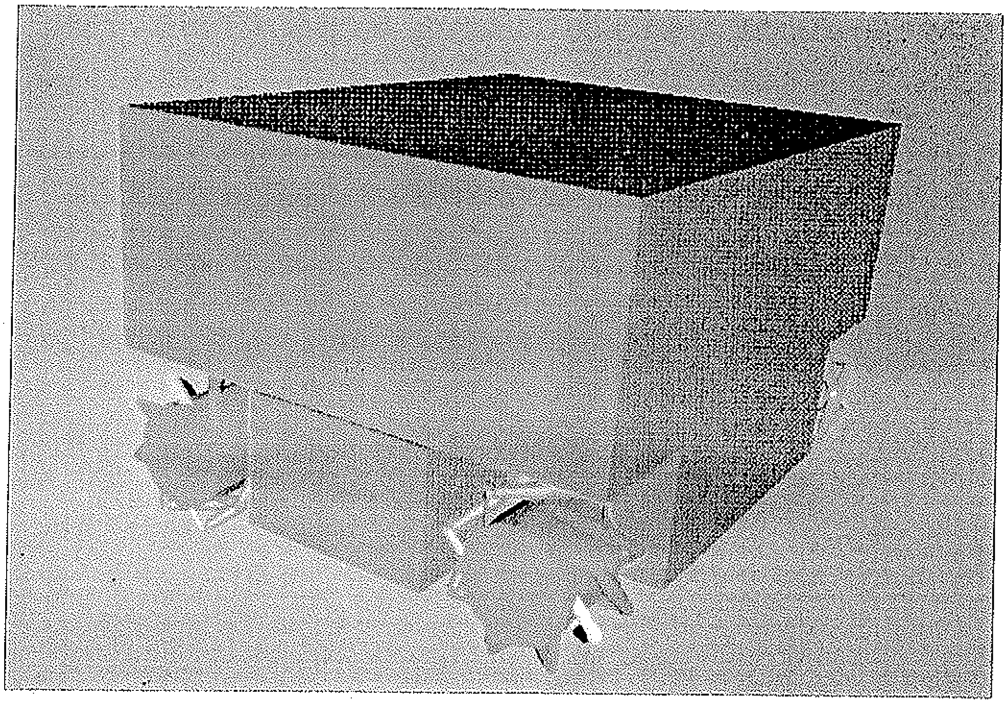

Figure 2: The geometric robot model.

reflection properties. The models can be exchanged to make use of other robots and environments.

Physical modelling: A simple model of robot motion on the basis of wheel velocity effects has been developed. A simple (ideal) and a more sophisticated simulation of ultrasonic sensing have been developed. These simulations can be coupled to the geometric sonar sensor model described before, and can be used to compute sonar distance measurements in the simulation at arbitrary sensor positions (cf. Fig. 4).

For a start, a raytracing-based model of light distribution has been developed to simulate the sensing behaviour of a light sensor. In order to make use of a more realistic model of illumination of different scenes, radiosity techniques can be incorporated. A simple raytracing model checks for collision detection within the simulation.

User Interface and Communication: Initially, the user has to specify the components of the robot, its control system and the simulation (environments and sensors). For this purpose, control panels and buttons have been introduced which provide a selection of already compiled system components. After selecting the desired modules, the corresponding files are linked together and a Unix shell script is created to launch all the necessary processes. The simulation and visualization are able to deal with multiple dynamic objects within the simulation, and multiple robot controllers can be defined to control these dynamic objects. Additionally, input devices can be associated with dynamic objects, hence enabling the user to interact with the simulated robot and the environment directly. Buttons and sliders are provided in order to switch be-

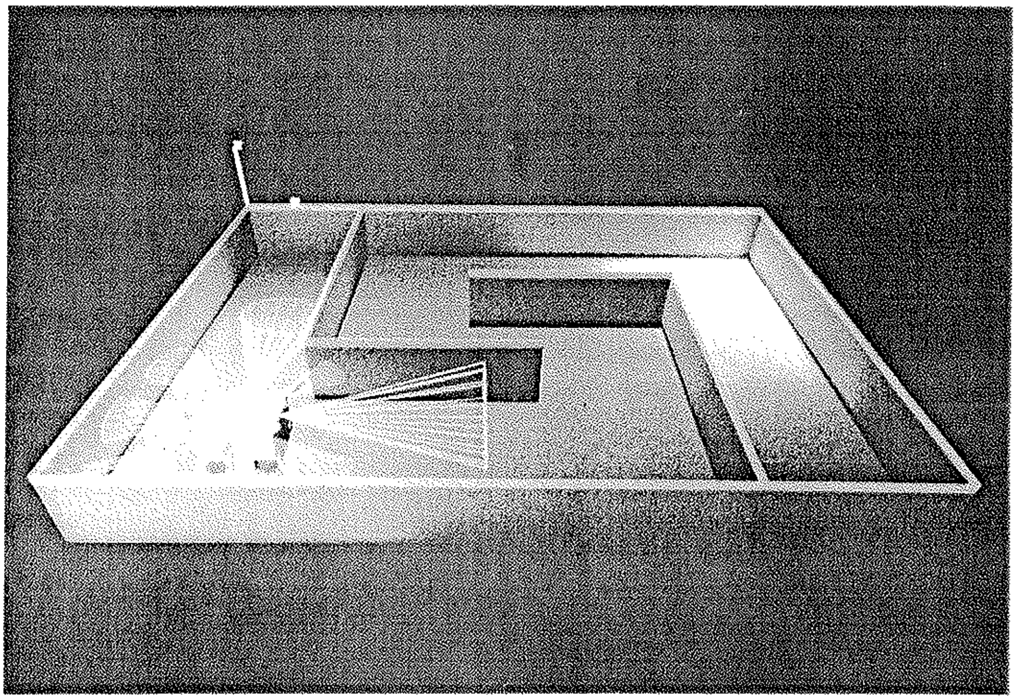

Figure 3: The user view of the maze from above.

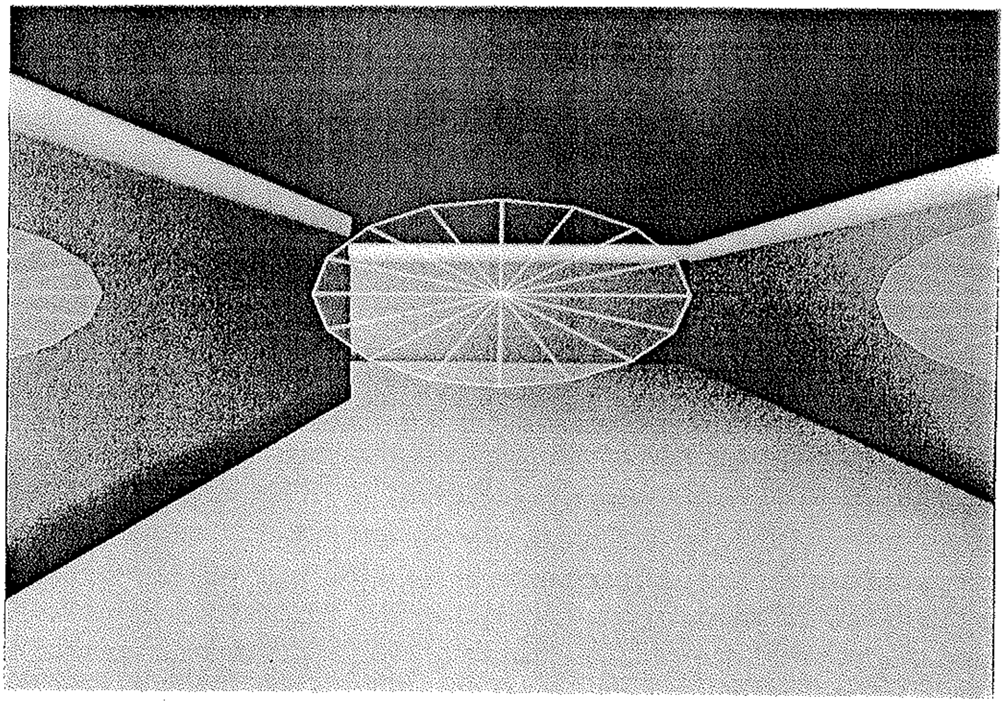

Figure 4: The user view from robot position.

tween the different views, and to inspect the ongoing robot controlling and learning processes.

For the user-robot communication the human supervisor needs to:

- know about the robot learning state (Merkwelt).
- be able to guide the robot.
- be able to provide sample situations (environments) to support particular aspects of the robot-world interaction.
- help the robot to resolve a deadlock situation.

It is intended to make the exploring and testing process as natural as possible by using 3D input and output devices as provided with VR technologies.

Learning robot control algorithms: The development of learning control algorithms is underway on the basis of a behaviour-based control approach. In our current approach, simple reflexes have been implemented to drive the robot through the simulated environment on the basis of (simulated or real) sensor signals. This way, raw sensor data can be transformed into motor commands quickly and reliably. A modularization is planned for different behavioural aspects such as wall following, crossing and room traversing, turning, and obstacle avoidance. More recent work focusses on the use of neural nets and learning classifier systems to install adaptive behaviour through a sequence of training sessions (cf. [3]).

5 Future Work

We expect that in the future powerful workstations can perform the radiosity computation and scene rendering on the basis of the light distribution data online. At the moment, a particular scene having a static light distribution can be displayed from various perspectives. One disadvantage, however, is the long compute time of radiosity data as well as the lack of dynamics in case the scene is changing. These aspects limit the application of radiosity techniques in Virtual Reality, where real-time performance is required (cf. Drucker et.al. [4]).

Hence, one focus will be on improving radiosity techniques with respect to efficiency and dynamics. A fast parallel algorithm for radiosity computation on a Connection Machine CM2 has been implemented and is used to prepare input for scene rendering on Silicon Graphics workstations. It would be helpful to have incremental radiosity techniques to allow real-time computation of dynamic light distributions. Also, future work will include the use of more advanced obstacle detection techniques developed on the CM2.

For the user-robot interaction the modules are under development. At the moment the user can start the robot at a position to be defined and watch its reaction when maneuvering through the maze. The future version will make use of a data glove and trackers to allow to interact with the ultrasonic sensors and to re-position the virtual robot (as this can also be done with the physical robot). At that stage, interactive modifications of the scene should provide new insights into the robot's learning process.

The visualization of the sensor data has been started for ultra-sonic data. The range of the ultra-sonic beam is displayed as a elliptical cone which marks the distance to an obstacle. Different colours are used to indicate the distance values, as well. Either simulated data or measured data can be displayed.

Further work will go into the modelling of the imaging of spheres to provide an intuitive way of displaying complex control and abstraction data. The visualization of the robot control data is in the discussion and definition phase. We think about enriching the display of robot control data by including information about the environment, e.g. the knowledge that the robot is located within a room. Man-made rooms are usually of rectangular shape and have lower and upper boundaries. This way, we can transform simple robot data in more intuitive display data and visualize it appropriately.

Acknowledgements

We would like to thank the head of the Scientific Visualization group at GMD, Wolfgang Krüger, for providing the chance to develop our subtle and early ideas into real project work. Thanks go also to our colleagues Christian-A. Bohn, Axel Dirksen and Jürgen Ziehm for their implementational support and help during the last year. Finally we would like to thank Ulrich Licht and Rüdiger Westermann for useful discussions through an earlier stage of the project.

References

[1] Foley, van Dam, Feiner, and Hughes. *Computer Graphics: Principles and Practice.* Addison Wesley, 1990, 2nd ed.

[2] U. Schnepf, A. Asteroth, and M. S. Fischer. Etholab - a System for the Configuration and Evaluation of Behavioural Synthesis in Autonomous Systems. In *Achtes Fachgespraech ueber Autonome Mobile Systeme*, November 1992. Karlsruhe, Germany.

[3] M. Dorigo and U. Schnepf. Genetics Based Machine Learning and Behaviour Based Robotics: a New Synthesis. *IEEE Transaction on Systems, Man, and Cybernetics*, 23(1):141–154, January/February 1993.

[4] S. Drucker and Peter Schroeder. *Fast Radiosity Using a Data Parallel Architecture.* Thinking Machines Corporation, Cambridge, 1991.