DrIVE: A VIRTUAL TRAINING ENVIRONMENT FOR DRIVING BEHAVIOUR

C. Diplas[1,2] D. Giakovis[1,2] A. Kameas[2] P. Pintelas[1,2]

cdplus@math.upatras.gr dg69@math.upatras.gr kameas@math.upatras.gr pintelas@math.upatras.gr

[1] Division of Computational Mathematics & Informatics, Department of Mathematics, University of Patras, Hellas
[2] Educational Software Development Laboratory, Department of Mathematics, University of Patras, Hellas

Abstract

A highly interactive computer-based environment where the user becomes part of an "artificially real" world, is defined as virtual Reality (VR). Conceptually, VR is the process which gives the user the possibility to be present into abstract spaces where the machine is transparent and the natural observer is absent. VR is closely related to the natural behavior. Up to date, there are applications of VR into a variety of domains, such as architectural walkthrough, teleoperation and teleconferencing, remote collaboration, medical and military simulations, etc. The education/training via VR is a new trend in international scale. In this paper, a virtual training environment (DrIVE) is presented.

DrIVE (**Dr**iving an **I**nteractive **V**irtual **E**xperience) is a virtual training environment intending to train car-drivers in common driving situations. It is based on desktop VR, with a minimum of requirements in processing power and data storage. DrIVE consists of three main parts: driving lessons, tests for the trainee, and free driving with on-line guidance. First part includes simple driving cases-lessons which are based on simulated situations with traffic lights, crossroad, traffic signs etc. A number of multiple points of view is provided, so the trainee can be either inside the cars which participate in the above mentioned situations or a human standing on the side of the road and so on. The second part includes a number of tests concerning the situations that the trainee has already been taught in the first part. DrIVE simulates crashes (with sound effects) in those cases that the results of the trainee answer is incorrect. In third part, the trainee becomes the driver of one of the cars and drives into the environment. In this part, a referee entity, invisible to the trainee, informs him on-line about his performance, warning him in cases that he falls into mistakes, or helps him about the correct car movement.

Although the environment is experimental, some user feedback concerning the use of VR as educational and training tool is already available and, in fact, extremely encouraging.

keywords: virtual environments, interaction, training

1. Introduction

A highly interactive computer-based environment where the user becomes part of an "artificially real" world, is defined as Virtual Reality (VR). Conceptually, VR is the process which gives the user the possibility to be present into abstract spaces where the machine is transparent and the natural observer is absent [1]. The computers, displays and interactive graphics technology evolution, combined with the construction of novel devices for human-computer interaction and the development of advanced algorithms for digital signal processing and image compression are leading to the development of a series of techniques, under the term Virtual Reality [2,3].

Virtual Reality extends the capabilities of Computer Assisted Instruction. Often, a virtual environment consists of emulations that go beyond the common man-machine interaction metaphors and include user's perception of being part of the simulated environment. User manipulates virtual environment objects and events in a manner that the typical computer simulations cannot support [4]. The big difference between a VR system and a single computer simulation is that VR extends the human nervous system and becomes a generator of reality, while computers simulation involve simple symbolic processing. VR achieves this at least at the interface and communication levels. User becomes part of the virtual world which, having its own properties, behaves as the real one [6].

VR is closely related to the natural behaviour. Programming, keyboard and mouse are substituted with physical operations such as hand gestures, movement and speech. Computer being an ideal tool for symbolic and abstract manipulation may provide the interface to this manipulation by adopting VR technology in order to train by experience. In this context, DrIVE is presented in this paper. In the following sections, an introduction to DrIVE virtual training environment will be presented. Next, design and interaction aspects are discussed

and the architecture of DrIVE is presented and some snapshots are depicted. Finally, the paper concludes with relevant further work in the domain.

2. DrIVE - Introduction

DrIVE is a virtual training environment intending to train novice car-drivers in common driving situations. It is based on desktop VR, with a minimum of requirements in processing power and data storage. DrIVE consists of three main parts: driving lessons, tests for the trainee and free driving and on-line guidance.

The driving lessons part is based on simulated "real world" driving situations. So, lessons include traffic lights situations, signs and simple priority cases. The trainee can watch the responses and attitude that an experienced car-driver demonstrates in similar situations.

A number of multiple point of views are provided by the environment. For example, the trainee can be inside any of the cars which participate in the above mentioned situations or can take the place of a human being standing in the side of the road and so on. This part is executed without the participation of the trainee as an active object of the environment, but he can replay or stop the lesson script.

A number of tests concerning the situations that the trainee has already been taught in the first part are included in test part. In critical moments as the cars, for instance, come close to a crossroad, a multiple choice question is displayed on the screen. These questions are about actions that a real driver could take in the same real world situation. The trainee can click on one of the answers provided. The system records all trainee responses for future analysis and trainee performance evaluation.

In third part, the trainee becomes a driver of one of the cars. He has the possibility to interact with the virtual environment's objects and the environment itself. In this part, a referee

entity, invisible to the trainee, informs him on-line about his performance, warning him in cases that he takes the wrong driving action, or helps him with the correct car movement.

3. Design and interaction aspects

DrIVE has been developed with Superscape VRT software [8] which is an Object Oriented development environment for virtual objects. Properties can be attached to objects which constitute the DrIVE environment using a number of editors (world editor, shape editor, texture editor, sound editor etc.), and behavioural characteristics are assigned to them using a C-like programming language, named SCL (Superscape Control Language). The code is executed once per frame, and the developer is able to associate an SCL code with each object in order to perform actions which cannot be performed by the properties that have been given to it using the above mentioned editors [8]. Code execution is event driven, but some common tasks are performed directly with the routine libraries that are provided by this language.

Most interactive objects of the DrIVE virtual environment use their own SCL code, in order to have some kind of "intelligence", and special tasks and behaviour are performed when the objects interact with each other or with the trainee. A referee entity, called control object, is responsible about the coordination of the whole virtual world and the trainee interactions.

During the design of DrIVE some standard user interface aspects such as dialog and message boxes were preserved. So, dialog boxes appear each time the trainee has to choose among a set of answers, or each time the system informs the trainee about the situations that may be met during the course. On-line messages are shown when trainee actions are wrong. Nevertheless, in the major part of the application, except the above mentioned interface standards, the interface is "transparent" and the user interacts directly with the virtual environment objects. The system is based on 3D-graphics running on a standard PC unit

with minimum requirements, and gives the user the ability to interact with the virtual environment and its objects using mouse and keyboard.

If someone adopts the meaning of the term "interactivity" as this introduced by Bowman and Hodges [7]:"...the user's ability to create, manipulate, or change the objects in the environment, or the environment itself", then DrIVE is characterized by a medium level of user interactivity.

The training domain of DrIVE, concerns an everyday task as the car driving is, so the training requirements were no difficult to de determined. In fact, these requirements were taken from National Driving Code Manual. On the other hand, it was not possible to cover the entire Driving Code, so the effort was on the role of virtual environments as a training methodology and mostly as experience transferring metaphor.

DrIVE is not a driving simulator but a training system which was created with desktop-VR techniques [6]. By this we mean that DrIVE is not going to train someone on car driving (e.g. changing gears, usage of lights) but to get the user to confront with some routine situations that happen during a "real world" everyday driving, for example to learn that he/she ought to give priority to cars coming from his/her right, to obey to traffic signs and/or lights, to drive the car on the correct side of the road, and have his/her attention when comes close to the pedestrian crossings.

4. System Architecture

4.1 First Part (Lessons)

In this part, the lesson scenario is described shortly, and the cars positions at the start of the lesson are shown.

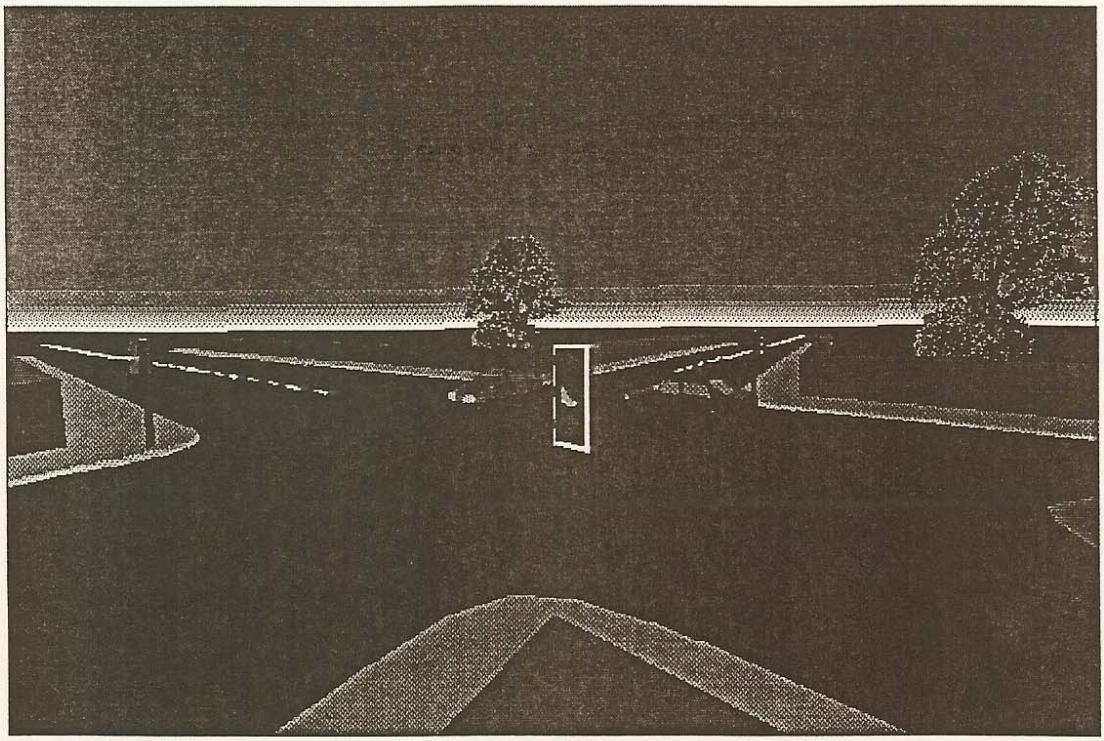

figure 2: The system has already gave its advice about the right order the cars must pass the crossroad, and now demonstrates the correct car routes to the trainee

The trainee watches the car route and the program informs him on the possible dangers during the "trip" and which road behaviour is valid or not (see picture above). Lessons are short and the trainee is able to choose among the lessons that are provided. Communication between the trainee and the system is done with messages which inform him/her about what is happening or is going to happen.

4.2 Second Part (Tests)

In second part, the trainee has to undergo an examination. One car is the "leading actor" of the test script, and this car obeys to the trainee's orders during DrIVE sessions. In this case, he has to answer a set of simple multiple choice tests. The "leading actor" car, meets a number of situations, each calls out the trainee to choose the correct answer, among a set of answers and decides about the correct car route. The system then informs him/her about the consequences of his/her choice, as the case may be. For example, in case the trainee decides

not to stop the car, but allows it to continue its course in front of a crossroad without signs and another car is coming to its right, the two cars crash, using pre-recorded "real world" crash sounds, as sound effects. If he/she chooses the correct answer, a man standing at the pavement, into the virtual environment of course, encourages the trainee to continue by announcing successful test pass using recorded speech and speech bubbles (see picture below).

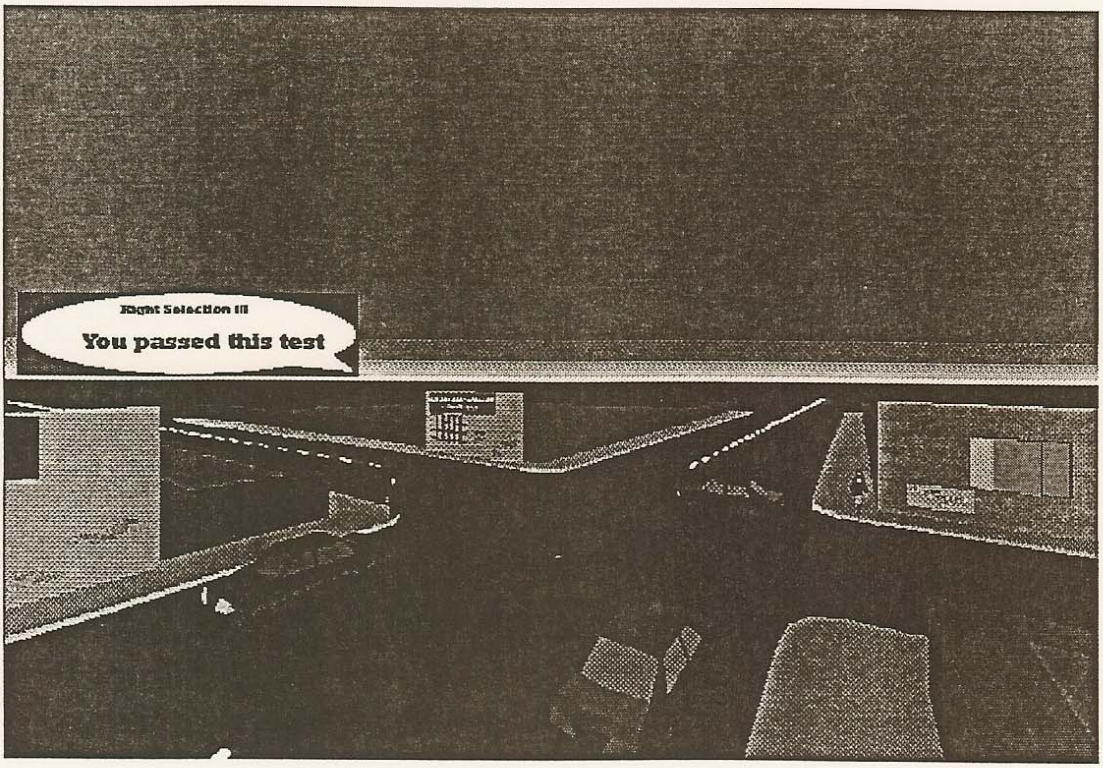

figure 2: The trainee has already passed this test, and the man standing on the pavement, under the guidance of the referee entity, approves him about his choice

These multiple choice tests provide an assessment of trainee performance, during the first part that described above by giving statistics (in the form of pie charts).

4.3 Third Part (Free driving)

In third part, the trainee himself/herself becomes the driver of the "leading actor" car and drives it using mouse or keyboard. Another input device (spaceball, dataglove, 3D-joystick

etc.) could be used instead. The system (specially the referee entity) keeps track of his performance and informs him/her on-line about the possible mistakes or faults he/she falls into (see picture below). Cases that already supported are: supervision about the motion of the car in the right side of the lane and the way the car turns (open or sharp bends).

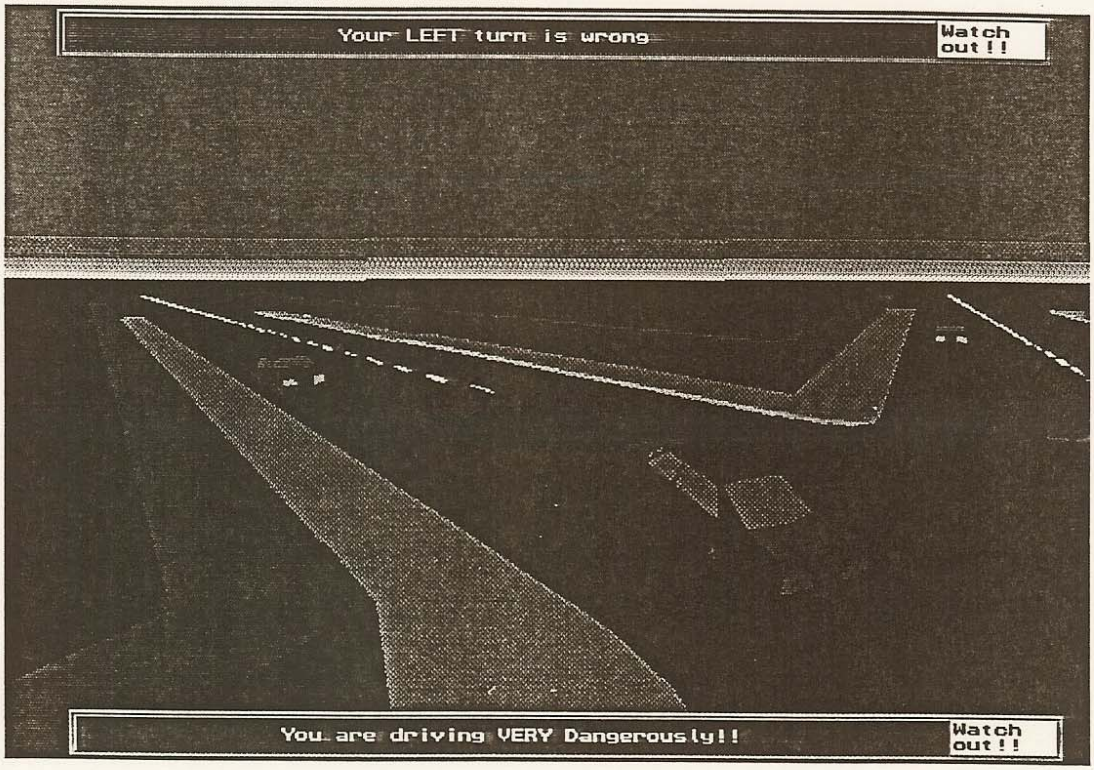

figure 3: The trainee is turning so close to the left side of the road, so the system informs him about it with on-line messages

This part is designed in a way that allows the future embodiment of "intelligent" characteristics into the "leading actor" and the other cars of the virtual environment. In this way, cars can have their own behaviour with a built-in driving code as a knowledge base, and some basic rules about the interaction with each other or with the trainee.

5. Conclusions

The DrIVE virtual environment presented here, is an experimental prototype and not a complete virtual training environment. The system created in order to evaluate the

capabilities of the virtual environments as computer-based training (CBT) tools [5,6]. The design and development team aspires to provide an integrated tutoring environment to novice drivers, related with the driving behaviour.

With DrIVE, a new approach to the training on driving code and driving behaviour is proposed. A novice driver can learn and practice on the correct driving behaviour whenever he wants to, in a safe simulated environment. Some common traffic situations are supported in order to give these people the opportunity to watch and practice on the correct car driving reactions.

The aim of the application is to transfer experience on this domain, which can be done using the synthetic experience that virtual environments are able to provide rather than actual practicing with real cars which involves obvious dangers. To this end, feedback from users of this first prototype is extremely encouraging and several ideas for its improvement have been suggested to the design team.

Thus, additional features will be included into the system such as more driving lessons and test cases, as well as a users' performance evaluation mechanism, making DrIVE more reliable and efficient. Furthermore, a full training and testing environment (microworld) will be provided which will include all kinds of driving conditions such as unexpected pedestrians, cyclists, snow storms etc. Finally, in a second step, artificial intelligence techniques (e.g. neural nets, intelligent agents) are going to be included, in an attempt to convert DrIVE into an Intelligent Tutoring System (ITS).

References

[1] Helsel, S.K. & Roth, J.P., eds (1991), "Virtual Reality: Theory, Practice, and Promise", Westport CT: Meckler Corporation

[2] C. Feiner and S. J. Gibbs, "Virtual Worlds". ACM Trans. On Information Systems (11-3), July 1993, editorial.

[3] J. Fuchs and G. Bishop, "Research Directions in Virtual Environments. An Invitational Workshop on the Future of Virtual Environments". TR92-027, March 1992, The University of North Carolina at Chapel Hill, Department of Computer Science.

[4] J. Sturman and D. Zeltzer, "A Design Method for "Whole-Hand" Human-Computer Interaction", ACM Trans. On Information Systems (11-3), July 1993, pp. 219-238.

[5] Meredith Bricken and Chris M. Byrne, "Summer Students In Virtual Reality: A Pilot Study On Educational Applications Of Virtual Reality Technology". Human Interface Technology Laboratory (HITL) of the Washington Technology Center (WTC) at the University of Washington (UW).

[6] A. Mikropoulos, C. Diplas, D. Giakovis, A. Halkidis, P. Pintelas, "Virtual Reality & Education: New Tool or New Methodology?", Proceedings of 2nd Conference on Informatics in Education, November, 11-13, 1994, Athens, Hellas.

[7] Doug A. Bowman, and Larry F. Hodges "User interface constraints for immersive virtual environment applications" ,Technical report 95-26, Graphics, Visualisation and Usability Center, Georgia Institute of Technology, USA, 1995.

[8] VRT 3.60, Superscape Ltd.