SACARI: An Immersive Remote Driving Interface for Autonomous Vehicles

Antoine Tarault, Patrick Bourdot, and Jean-Marc Vézien

LIMSI-CNRS, Bâtiments 508 et 502bis, Université de Paris-Sud,
91403 Orsay, France
`{tarault,bourdot,vezien}@limsi.fr`
`http://www.limsi.fr/venise/`

Abstract. Designing a remote driving interface is a really complex problem. Numerous steps must be validated and prepared for the interface to be robust, efficient, and easy to use. We have designed different parts of this interface: the architecture of the remote driving, the mixed reality rendering part, and a simulator to test the interface. The remote driving interface is called SACARI (Supervision of an Autonomous Car by an Augmented Reality Interface) and is mainly working with an autonomous car developed by the IEF lab.

1 Introduction

The aim of the project is to develop a Mixed Reality system for the driving assistance of an autonomous car. The applications of such a system are mainly teleoperation and management of vehicles fleet. To realize such an application, we have an immersive device called MUSE (Multi-User Stereoscopic Environment) (see Fig. 1), in the LIMSI-CNRS, and an autonomous car, PiCar [1], developed by the IEF lab.

Two major concepts were used in this project: telerobotics, and telepresence. Telerobotics is a form of teleoperation in which a human acts in an intermittent way with the robot [2]. He communicates information (on goals, plans…) and receives others (on realizations, difficulties, sensor data…). The aim of telepresence is to catch enough information on the robot and its environment to be communicated to the human operator in such a way that he should feel physically present on the site [3].

We took two existing interfaces as a starting point. In [4], Fong and al. define a collaborative control between the vehicle and the user. Queries are send to the robot, which executes them or not, depending on the situation. The robot can also send queries to the user, who can take them into account. This system can be adapted to the level of expertise of the user. The depth parameter of the scene is given by a multisensor fusion of a ladar, monochrome camera, a stereovision system, an ultrasonic sonar, and an odometer. They have developed two interesting driving interface: “gesture driver”, allows the user to control the vehicle with a series of gesture. Unfortunately, this driving method is too tiring for long distances. PDAdriver, enables to drive a robot with a PDA. In [5], McGreevy describes a virtual reality interface for efficient remote driving. His goal was to create an explorer-environment interface instead of a classical computer-user interface. All the operations, objects, and contexts must be comparable to those met in a natural environment.

V.S. Sunderam et al. (Eds.): ICCS 2005, LNCS 3515, pp. 339 – 342, 2005.

Fig. 1. MUSE immersive device

2 Global Architecture of the System

To answer the time and situational consciousness constraints of a telepresence system, we had to prefer the transmission of video data from PiCar to a virtual reconstruction of its environment. SACARI must also send/receive data for/from the wheel orientation and speed actuators/sensors. We transmit video and orders on two separate channels. PiCar controlling station is connected to the immersive device by a gigabit network between the IEF lab and the LIMSI-CNRS (see Fig. 2).

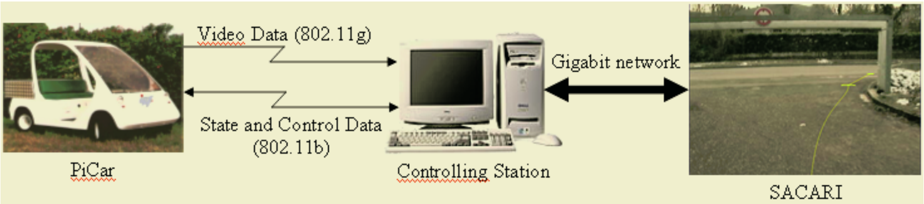

Fig. 2. Architecture of the system

To fulfill the telepresence constraints, we chose to make a system with short time delay, distributed operations, and modular components. We also needed software for data exchange, and the possibility to record/replay data from different tests. SACARI and PiCar's software part are developed on the same platform addressing these constraints: RTMaps (http://www.intempora.com). Each module is represented as a *Component*. We have developed three main groups of components for our application: a devices driver component, a rendering component, and a vehicle behavior simulator.

3 A Scene Graph-Based Renderer

The rendering Components have several constraints. First, it has to be able to simulate PiCar's environment. It should integrate video texture and complex virtual objects. The renderer library must be easily changed and inserted in a multithreaded application (RTMaps). Finally, the system should support multiscreen display, as our immersive device is biplan. The use of two screens is primordial. The telepresence recommends that the user feels present on site. That's why most of his field of view must be filled by the virtual scene. Moreover, we have noticed that a single screen limits the range of action of a user, especially when he wants to change the vehicle's direction using a 6 DOF tracker: he tends to give a direction only lying in the vision range. These constraints made us choose OpenSceneGraph (www.openscenegraph.org).

We have developed several kinds of "OSG Components" in RTMaps: Transformations, graphical objects which represent the leaves of the scene graph, video objects that can read RTMaps-produced image flow, and a viewer. This component can be set to drive a cluster for graphical rendering, switch between different kinds of navigation, specify graphical and stereo settings.

4 The Car Simulator

The device allowing the user to control the vehicle should work in semi-autonomous *and* non-autonomous modes. That's why we chose to use a 6 DOF Tracker to manage vehicle remote driving instead of a classical steering wheel. To drive the tracker, we use our own devices drivers manager: VEserver [6]. It is a real time client/server devices drivers manager with a distributed architecture that can drive synchronously numerous devices. We have integrated a client of the VEserver in RTMaps. An extern VEserver node tracks the events coming from an ARTTrack wireless 6DOF tracker. The vehicle is driven as presented in Fig. 3.

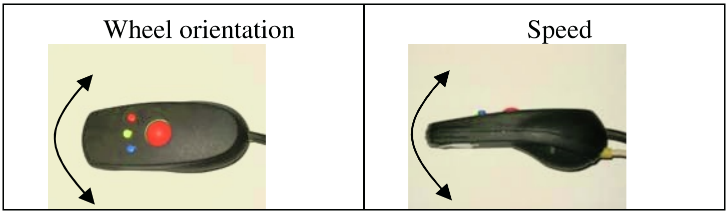

Fig. 3. Using a 6 DOF Tracker to drive the vehicle

We developed a navigator transforming the speed and the wheel orientation, given by the 6 DOF tracker, into a position and orientation of the scene graph camera. Given the last camera orientation ψ, the speed *v* and the front steering β, the back steering α and the length *L* between the nose gear wheels and the aft wheels, we can calculate the differential position and orientation of the vehicle:

$$\dot{x} = v \cdot \cos\psi \qquad \dot{y} = v \cdot \sin\psi$$

$$\dot{\psi} = \frac{v}{l}\tan\beta \qquad \text{with} \quad l = \frac{\alpha L}{1+\alpha}$$

Then, we integrated the autonomous car simulator, realized by the IEF lab, into our system. The component we developed takes the wanted position and orientation in input and gives the trajectory, the speed and the wheel orientation on output. The display shows the calculated trajectory and the next wanted point to reach (see Fig. 4).

Fig. 4. Control of the autonomous vehicle

5 Conclusion and Perspectives

We have developed all the needed tools for a remote driving application:

- an easy-to-use scene graph descriptor, which can be reused for VR applications,
- A simulator to test the different ways to control PiCar
- An interface dedicated to the remote driving and supervision of the vehicle

The next step will be to test our interface in real conditions, for remote driving and supervision. Another awkward point will be the transition from supervision to remote driving of the vehicle. It must be as natural as possible for the user. We plan to test different devices to perform such a transition.

References

1. S. Bouaziz, M. Fan, A. Lambert, T. Maurin, R. Reynaud, "PICAR: experimental Platform for road tracking Applications", *IEEE IV2003 Intelligent Vehicle Symposium,* ISBN 0-7803-7848-2, pp. 495-499, Columbus, June 2003
2. H. S. Tan, K. H. Lee, K. K. Kwong, "VR Telerobot System", *Proceedings of the 5th International Conference on Manufacturing technology,* Beijing, November 1999
3. B. Hine, P. Hontalas, T. Fong, L. Piguet, E. Nygren, "VEVI : A Virtual Environment Teleoperations Interface for Planetary Exploration", *SAE 25th International Conference on Environmental Systems*, San Diego, July 1995
4. T. Fong, C. Thorpe, C. Baur, "Advanced interfaces for Vehicle Teleoperation: Collaborative Control, Sensor Fusion Displays, and Remote Driving Tools", *Autonomous Robots 11*, pp. 77-85, 2001
5. M. W. McGreevy, "Virtual Reality and Planetary Exploration", *A. Wexelblat (Ed.), Virtual Reality: Application and Explorations*, pp. 163-167, 1993
6. D. Touraine, P. Bourdot, Y. Bellick, L. Bolot, "A framework to manage multimodal fusion of events for advanced interactions within Virtual Environments", *8th Eurographics Workshop on Virtual Environment*, 2002