The Simulation of a Billiard Game Using a Haptic Interface

Lucio Tommaso De Paolis
Dept. of Innovation Engineering, Salento University, Italy
SPACI Consortium, Italy
CACT/NNL, Italy

Marco Pulimeno
CACT/NNL, Italy

Giovanni Aloisio
Dept. of Innovation Engineering, Salento University, Italy
SPACI Consortium, Italy
CACT/NNL, Italy

Abstract

Recently the computer entertainment technology has generated a deal of interest among researchers and developers as it is recognized as showing high promise in creating exciting new forms of human computer interaction. Performance improvements in graphics hardware and the diffusion of the low cost haptic interfaces have made it possible to visualize complex virtual environments and provided opportunities to interact with these in a more realistic way.

In this paper a Virtual Reality application of a game of billiard game is presented; to allow to the user an interactive and realistic interaction is provided a force feedback by means of a commercial haptic interface. To build an immersive virtual environment has been used the development environment XVR and to simulate the rigid body dynamics has been utilized the ODE library.

1. Introduction

The field of computer entertainment technology has aroused great interest recently among researchers and developers in both academic and industrial fields as it is recognized as showing promise in terms of generating exciting new forms of human computer interaction.

Techniques used in computer entertainment are also seen to translate into advances in research work ranging from industrial training, collaborative work, novel interfaces, novel multimedia, network computing and ubiquitous computing.

At the same time, the performance improvements in graphics hardware and the diffusion of the low cost haptic interfaces have made possible the visualization of complex virtual environments and the opportunity to interact with these in a more realistic way. Haptic feedback in virtual environments makes it possible to increase the overall realism of a simulation by improving the user experience.

Haptic interfaces in virtual environments have been intensively studied in the past decade. Different types of haptic interface are used in virtual games which provide multimodal feedback creating a deeper sense of ”being” in control of the game.

Vibration is a practical and important way of providing haptic feedback to users. It requires a minimal interface design and has low complexity and cost. Players like to get some bodily feedback, be it vibration, movement or other. Vibration feedback joysticks are widely used input devices in current games.

2. Used Technologies

2.1. Phantom Omni Haptic Interface

The haptic interface used in this simulation is the PHANTOM Omni of SensAble Technologies; the device offers 6 degrees of freedom output capabilities.

2.2. eXtreme Virtual Reality (XVR)

XVR is an integrated development environment for the rapid development of Virtual Reality applications. Using a modular architecture and a VR-oriented scripting language, XVR content can be embedded on a variety of container applications making it suitable to write content ranging from web-oriented presentation to more complex VR installations.

The execution environment, the web browser plug-in and the virtual machine for this system have been developed by PERCRO Laboratory of Scuola Sant’Anna of Pisa, Italy and the XVR platform has

1550-6525/07 $25.00 © 2007 IEEE
DOI 10.1109/DS-RT.2007.30

been used in many virtual reality projects.

Originally created for the development of web-enabled virtual reality applications, XVR has evolved in recent years into all-around technology for interactive applications. Beside web3D content management, XVR now supports a wide range of VR devices (such as trackers, 3D mice, motion capture devices, stereo projection systems and HMDs) and uses a state-of-the-art graphics engine for the real-time visualization of complex three-dimensional models that is perfectly adequate even for advanced off-line VR installations.

XVR applications are developed using a dedicated scripting language whose constructs and commands are targeted to VR, and give the possibility to developers to deal with 3D animation, positional sounds effects, audio and video streaming and user interaction.

In its current form XVR is an ActiveX component running on the various Windows platforms, and can be embedded in several container applications including the web browser Internet Explorer.

XVR is actually divided into two main modules: the ActiveX Control module, which hosts the very basic components of the technology, like the versioning check and the plug-in interfaces, and the XVR Virtual Machine (VM) module, which contains the core of the technology, such as the 3D Graphics engine, the Multimedia engine and all the software modules managing the other built-in XVR features.

It is also possible to load additional modules which offer advanced functionalities, like the support to VR devices, as decision was made to keep them separated so that web applications, which usually do not need any of these advanced features, are not afflicted by additional downloading times.

The modularity of XVR makes it easily adaptable it both for web and for stand-alone applications, according to the specific needs, as it is made up of a central core (the XVR-VM) and several additional modules, the main being the ActiveX Control for the web access [1].

2.3. Open Dynamics Engine (ODE)

The Open Dynamics Engine is an open source, high performance library for simulating rigid body dynamics. It is a fully featured, stable, mature and independent platform with an easy to use C/C++ API. It has advanced joint types and integrated collision detection with friction. ODE is useful for simulating vehicles, objects in virtual reality environments and virtual creatures. It is currently used in many computer games, 3D authoring tools and simulation tools.

ODE is a free, industrial quality library for simulating articulated rigid body dynamics. It is good for simulating ground vehicles, legged creatures, and moving objects in VR environments.

ODE is good for simulating articulated rigid body structures; an articulated structure is created when rigid bodies of various shapes are connected together with joints of various kinds.

ODE is designed to be used in interactive or real-time simulation and it is particularly good for simulating moving objects in changeable virtual reality environments.

The ODE collision system provides fast identification of potentially intersecting objects and a non-penetration constraint is used whenever two bodies collide; the current collision primitives are sphere, box, capped cylinder, plane, ray, and triangular mesh. However it can be ignored and an alternative collision detection can be used [2].

2.4. OpenHaptics

The SensAble OpenHaptics toolkit enables software developers to add haptics and true 3D navigation to a broad range of applications, from design to games and entertainment to simulation and visualization.

Using the OpenHaptics toolkit, developers can leverage the existing OpenGL code for specifying geometry, and supplement it with OpenHaptics commands to simulate haptic material properties such as friction and stiffness.

The architecture enables developers to add functionality to support new types of shapes and it is also designed to integrate third-party libraries such as physics/dynamics and collision detection engines.

The OpenHaptics toolkit includes the Haptic Device API (HDAPI), the Haptic Library API (HLAPI), utilities, PHANTOM Device Drivers (PDD), and source code examples [3].

3. Model description

In the application it is possible to distinguish three different models: the graphical modelling, the physical modelling and the haptic modelling. Each modelling can be represented by a loop executed at a specific frequency; the XVR application combines all the loops.

3.1. Graphical modelling

The graphical model consists of a set of 3D objects modelled using 3D Studio and imported into the XVR development environment where they are managed using the XVR scenegraph.

The modelled objects are: the billiard table, the cue, the billiard balls and the skittle. An example of the snooker with five skittles has been implemented.

XVR provides classes for the lighting, shading and observation point (virtual camera) management; in addition it allows the user to superimpose 2D text on the scene and this functionality is used in order to provide the user with to the user messages about the system conditions, working modality and error states. Three different working modalities are allowed:

- *camera modality* to choose the desired position of the virtual camera with the end-effector mapped the cursor and no force feedback is provided;
- *play modality* when the end-effector is mapped on the cue and a force feedback is provided to the user by means of the haptic interface;
- *positioning modality* when a force feedback is provided for the user in order to allow the right positioning of the billiard balls in case of necessity or to decide the starting state of the game.

Since in the real game it is possible to use the left hand when aiming and striking the ball, in the play modality it is possible to fix the cue movement in the desired direction in order to allow a more careful aim and a more stable interaction in the virtual environment. A game situation is shown in Figure 1.

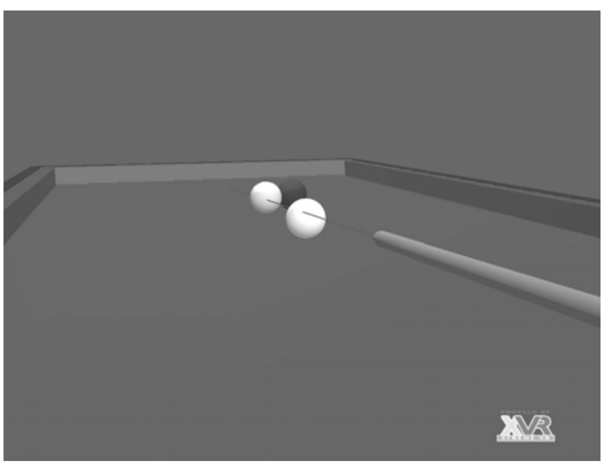

Figure 1. A game situation

3.2. Physical modelling

Each object of the scenegraph is modelled from the physical point of view defining the geometry, the mass, the inertia, the stiffness and the contact friction with another one.

The physical modelling definition is carried out using the functions provided by the library of the rigid bodies dynamic simulation (ODE); this library is also used to define the billiard game dynamics simulation.

The process of simulation of the rigid body system through is called integration. Each integration step advances the current time by a given step size, adjusting the state of all rigid bodies for the new time value.

The ODE integrator is very stable, but not particularly accurate unless the step size is small. Between each integrator step the user can call functions to apply forces to the rigid body and the sum of these forces will be applied to the body when the next integrator step happens.

At each simulation step, ODE is used to check the collisions between objects and to calculate the forces of interaction and those applied by the user; in addition the speed is computed and new orientation and positioning of the objects. In this way the state of the system is updated and the new one is provided.

3.3. Haptic modelling

Regarding the haptic modelling of the objects that people the virtual scene, two different solutions have been considered:

- the utilization of the HapticWeb library of XVR;
- the utilization of the OpenHaptics library provided with the PHANTOM Omni haptic device.

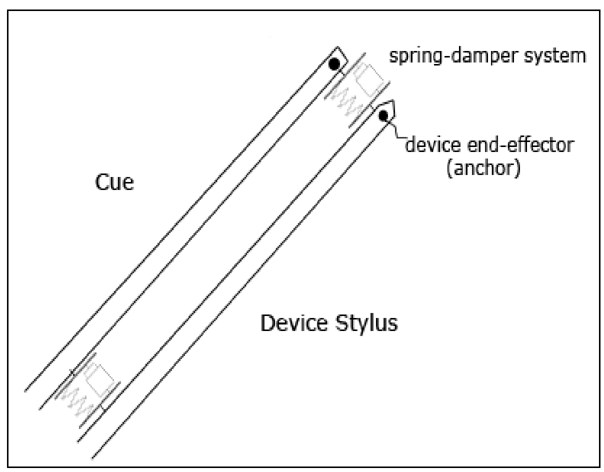

Figure 2. Virtual coupling for the cue movement

The second solution has been chosen in our application because it permit control at a lower level of the haptic interface.

The cue is modelled as a rigid body and, in the play modality, its position and orientation are constrained, using a spring-damper system, to the position and orientation of the stylus of the haptic interface; at each simulation step the forces depending of the stylus movement are applied to the cue in order to obtain a replication as soon as possible, while taking care to avoid stylus displacement. This model is shown in Figure 2.

The force feedback is provided by means of the haptic interface using the functionalities of the OpenHaptics library.

Another spring-damper system is virtually located between the end of the cue and the real position of the end-effector; the force exercised on the end-effector is sent to the haptic interface and provided to the user.

In this way the forces exercised on the cue by the colliding objects are sent to the user by means of the virtual coupling cue-stylus providing the force feedback due to the impact between the cue and the other objects.

Figure 3. A game phase

From the haptic point of view, the scene is rendered in two different ways: the play modality and the positioning modality.

In the play modality, the virtual coupling between the end-effector of the haptic device and the cue provides the force feedback to be sent to the user by means of the haptic device.

Figure 3 shows a game phase using the Phantom haptic device.

In the positioning modality, used to allow the right positioning of the balls in case of necessity or the decide the starting state of the game, the balls in the scene are attracted by the cursor (end-effector of the haptic device) and in this way the ball selection is easy.

4. Conclusion and future work

In this paper a Virtual Reality application of the billiard game is presented; in order to provide the user with an interactive and realistic interaction a force feedback is provided by means of the Phantom Omni haptic interface.

The virtual environment has been built using the development environment XVR and the rigid body dynamics have been simulated utilizing the ODE library.

An improvement of the immersion sensation could be obtained using a pair of the shutter glasses which are able to create the illusion of a stereoscopic image displaying an alternate frame sequencing for each eye.

10. References

[1] VRMedia, http://www.vrmedia.it.

[2] Open Dynamics Engine, http://www.ode.org.

[3] SensAble Technologies, Inc, http://www.sensable.com.