VIRGY: A VIRTUAL REALITY AND FORCE FEEDBACK BASED ENDOSCOPIC SURGERY SIMULATOR

Dr. Charles Baur (Lab manager) (baur@epfl.ch)
Didier Guzzoni (Research assistant) (guzzoni@epfl.ch)
Olivier Georg (Research assistant) (ogeorg@epfl.ch)
VRAI Group
IMT / DMT
EPFL (Swiss Federal Institute of Technology)
1015 Lausanne
Switzerland

Abstract

This paper describes the VIRGY project at the VRAI Group (Virtual Reality and Active Interface), Swiss Federal Institute of Technology (Lausanne, Switzerland). Since 1994, we have been investigating a variety of virtual-reality based methods for simulating laparoscopic surgery procedures. Our goal is to develop an endoscopic surgical training tool which realistically simulates the interactions between one or more surgical instruments and gastrointestinal organs. To support real-time interaction and manipulation between instruments and organs, we have developed several novel graphic simulation techniques. In particular, we are using live video texturing to achieve dynamic effects such as bleeding or vaporization of fatty tissues. Special texture manipulations allows us to generate pulsing objects while minimizing processor load. Additionally, we have created a new surface deformation algorithm which enables real-time deformations under external constraints. Lastly, we have developed a new 3D object definition which allows us to perform operations such as total or partial object cuttings, as well as to selectively render objects with different levels of detail. To provide realistic physical simulation of the forces and torques on surgical instruments encountered during an operation, we have also designed a new haptic device dedicated to endososcopic surgery constraints. We are using special interpolation and extrapolation techniques to integrate our 25 Hz visual simulation with the 300 Hz feedback required for realistic tactile interaction.

The fully VIRGY simulator has been tested by surgeons and the quality of both our visual and haptic simulation has been judged sufficient for training basic surgery gestures.

1. Introduction

The project aims at creating a training system for endoscopic surgery. The system is made out of two complementary axes.

The first one consist in developing a virtual representation of given organs (the focus is put on intestines) and in designing simultaneously a “surgical ghost” that would provide the surgeon with a force feedback through the endoscopic tools he manipulates. Such a feedback, added to the visual one performed through virtual reality, is mandatory to define a learning tool sufficently realistic to be usable.

The second axis consists in using a multimedia communication platform to allow both computer supported cooperative work (CSCW) and tele-teaching. CSCW will provide surgeons with the possibility to cooperatively work on a given surgical pathology irrespectively of their physical location. As far as teaching is concerned, students will be allowed to follow in all details, an endoscopic surgical operation. They will also be given the opportunity to interact with a (distant) educator by sharing the same virtual surgical scene on which they are learning. In short the multimedia communication platform allows to network the know-how and gestual skills within a community of surgeons and makes them available to students.

This whole design relies on real time computer graphics and data exchange as well as human interface technologies associated with virtual reality. The goal is to apply the capabilities of virtual environment combined with a priori knowledge stored on different mediums to improve, at several levels, endoscopic surgery techniques. Since this is a fairly new combination of technologies and a new application, our project is drawn from the current generation of surgical technologies and, by analogy, to other fields where virtual reality technology has been more intensively applied and studied.

2. Purpose

Endoscopic surgery is one of the revolutions that have deeply changed the surgery. The advantages are obvious. The surgery trauma as well as post-operatory pains and/or complications are drastically reduced. The time the patient has to stay in hospital is also shortened.

The fact that endoscopic surgery is considered as non invasive led however to several problems. The conventional senses (essentially visual and tactile) surgeons have through direct access and contact when performing a conventional operation are not available anymore. A correct use of the long and fine camera called endoscope requires a new learning process to be able to cope with new kinds of force and visual feelings.

The research described here will have a strong impact for surgeons. Thanks to the virtual based system we are developing, they will be able to learn how to use an endoscopic tool. Such a learning phase including both force and visual feedbacks similar to the ones they will meet in real cases, will allow a gesture and procedure optimizations without any danger for the patients and/or ethical or political limitations. Due to daily stronger regulations concerning both human and animal protections, and due to the lack of other efficient learning systems, such an approach has a tremendous scholar and industrial potential. The virtual environment will also enable an experimented surgeon to practice new strategies or compare approaches proposed by colleagues. Local and distant collaboration will be fueled by the integration of the virtual reality environment with the appropriate multimedia communication platform.

This project led in cooperation with surgeons aims at anticipating the new techniques that could help surgeons to improve the quality and the efficiency of minimal invasive surgery. To summarize, we can write the success of this project could lead toward safer and cheaper surgery without requiring large investments.

3. Methods

3.1 Virtual reality

3.1.1 The object based framework

Our library *(libptk v2.0)* is built over a clear and flexible object based hierarchy. The first goal is to separate the graphic part from the pure computing section. For instance, to deform a solid, the programmer does not have to deal with specific computer graphics instructions (such as OpenGL), but only with a set points and triangles stored in memory. The system is made of a collection of independent objects, smart enough to include all the parameters needed to be part of the simulation. Another important feature is that the scene to simulate is described in a file which is loaded at initialization time. So, even if the focus is put on virtual surgery simulations, it is possible to use the same software to run various types of scenes, such as monitoring and controlling a team of remote mobile robots.

As said before, each component of the simulation (rendered solids, textures, deformation modules, sensors, etc.) inherit a set of basic features from the same root class. For instance, they are able to react to external events, to load/store themselves from/into a file, provide a control panel. Moreover, once they are created they register to the world and so know each other and therefore exchange messages. A sensor know the solid it controls, a texture knows the topography of the object it should be applied to or a deformation context points the set mesh it has to modify.

3.1.2 The sensor engine and the agent architecture

Since a good simulator has to deal with users as intuitively as possible, we need to accept various devices in a flexible and robust way. *Libptk 2.0* allows to link any rendered solid (or the view point) to set of local (or distant) sensors such as force feedback devices or 3D mice. At each frame, the *sensor engine* checks all the sensor objects of the world asking them to update their values. Once the new values are read, the position of the targeted graphic object is updated. When a solid is linked to a set of sensors, all of them are read and the final value is computed by summing their values according to their weight.

An important feature is that sensors can be hooked up to the external world in two kinds of ways. The first one is local communication ports (serial, parallel, shared memory, etc.). The second way is to connect the sensor through the Open Agent Architecture (OAA)™ ☯1☸. This powerful feature allows us to share sensors (through network links) among

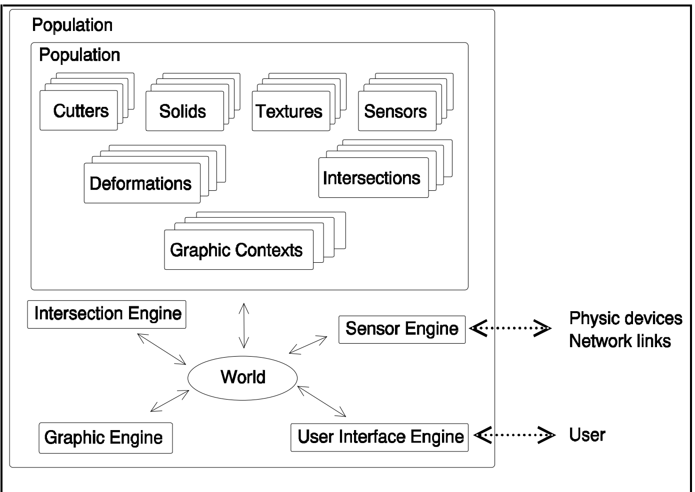

Figure 1: The libptk 2.0 framework.

various instances of libptk running on different locations at the same time. So surgeons can work cooperatively to prepare and operation or a teacher can remotely monitor a student and show her or him, in real time, the right gesture. Since the OAA is an ongoing project at SRI International, we can easily reuse part of their existing agents. For instance, as implemented for a few demos, we can use a speech recognizer as sensor, so the user can talk to the machine to set the position of a solid or the view point.

The following devices are currently available as libptk sensors (local and through the OAA):

- 3D mice : Space Mouse, Magellan (Teacher – Student interface)
- Force feedback devices : Pantoscope (EPFL), Immersion (Impulse engine) (Haptic interface)
- Computer pen : Wacom (Teacher – Student interface)
- Strereographic 3D vision system

3.1.3 Graphic contexts and animated texture mapping

A *graphic context* object regroups the information on how the solid it targets will be rendered. This includes colors, light effects and textures. All these data are set when loading an initial description file, but can also be modified afterwards on the fly through a control panel.

Among other features, the way a texture is applied to a solid can be modified, which allows the hiding of singularity points. Also several versions of a texture (called *phases*) can be mapped in turn onto an solid. Since the mesh is not modified, this way of animating the texture gives a visual feeling of pulsation of the solid, but not a tactile one. This is not irrealistic, since in an endoscopic approach, most of the feelings of the pulsations can only be visualized but not sensed through the tools. The proposed approach is therefore very efficient and doesn't cost any CPU time due to the use of dedicated hardware.

3.1.4 Real time deformations and cuttings

Among the object types provided by our library, there are deformable solids. Such a solid is deformed when touched by a set of segments attached to another solid. For instance, a surgical tool can carry segments and, therefore, it is possible to be notified when it touches some other solid within the scene.

Real-time deformation is achieved by letting the *intersection engine* search at each frame and for each set of segment (which models surgical tools) whether there is an intersection with a dynamic (or deformable) solid. If such is the case, it sends a message to a *reaction object* which will move the points of that targeted dynamic solid. The way a solid is deformed can be modified in several ways, like the extend of the deformation, the time of relaxation, the general shape of the deformation and how the solid reacts to a tool touching it non-perpendicularly. To be obtain real time behaviors, the shape of a deformation is computed by using a 3D function called a *profile*. Starting from the point of contact, the deformation it propagated from neighbor to neighbor according to the chosen 3D function.

The longer the distance from the intersection point, the less significant the effect of the deformation affects a given point. All the needed parameters are specified in a *deformation context object* and can be loaded at initialization and set on the fly to tune the deformation behavior independently for each organ.

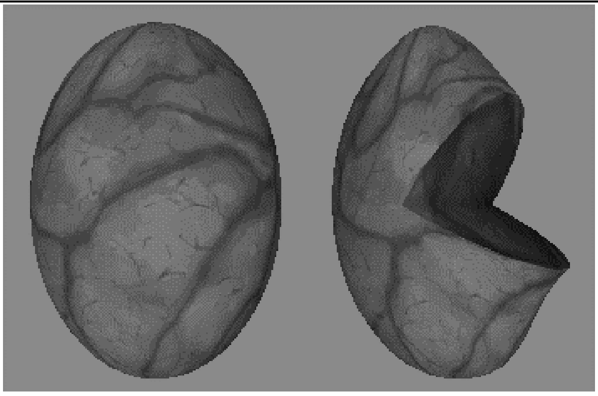

Figure 3: Before and after a cut.

3.2 Force feedback

Based on an investigation of minimally invasive surgical procedures the most stringent requirements for the force reflecting system have been determined. They can be traced back to two fundamental demands expressed by the surgeons: realistic force reflection and realistic visual appearance. Realistic force reflection implies a mechanical setup which provides for sufficient displacement ranges and output force capability in four degrees of freedom. In order to achieve realistic visual appearance, all mechanisms providing haptic feedback need to be completely hidden below the skin of the artificial patient. These specifications, together with the constraints related to the capabilities of the human sensori-motor system, have served as guidelines for the design of a novel hybrid parallel-serial spherical remote-center-of-motion force reflecting mechanism, named *PantoScope*☯2☸.

For virtual object simulations, a simple and robust *implicit force control scheme* including active compensation of frictional and gravitational effects is employed. The control law is based on the specification of stiffness and damping gain matrices which restrict the object force models to linear, time-varying functions of position and velocity. Underlying joint torque control loops that are closed around the motors and transmissions

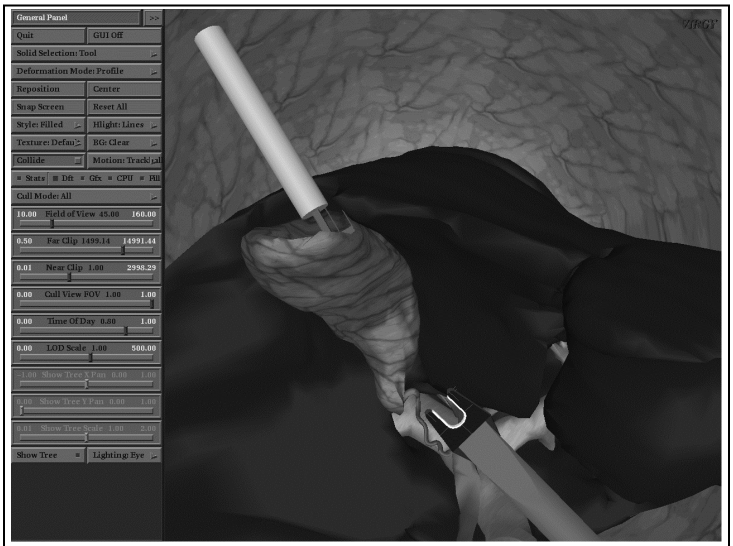

Figure 2: Real-time deformation of an organ

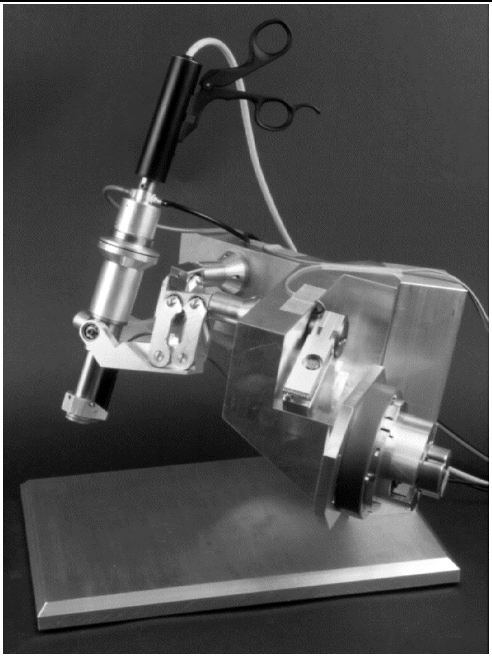

Figure 4: The PantoScope

are used to reduce the apparent friction in the actuator-transmission stages to approximately one hundredth of sustained maximum output torques and forces. The compensation of gravitational effects introduced by the weight of the force feedback manipulator's mechanical structure is achieved by adding feed-forward compensation torques to the commanded actuator torques. In this way, static and dynamic object attributes including weight, elasticity, and viscosity can be simulated over a wide dynamic range.

For the connection of the haptic feedback system to the object and organ impedance models computed on a remote graphic workstation, a *supervisory control scheme* is proposed. This control concept allows to deal with the fundamentally different bandwidth requirements of the human haptic and visual sensory systems and features particularly high robustness with respect to delays in the organ impedance model computations. Collision detection and impedance computations can be performed at a low and time-varying frequency at least one order of magnitude less than the update rate of the implicit force control loop without jeopardizing system stability.

4. Results

For the force feedback part of the project, the realized device features compactness, high sustained output force capability, low apparent inertia, low friction, high structural stiffness, zero backlash, and an absence of mechanical singularities within its large workspace. Finally, its virtual pivot allows to completely hide the force feedback manipulator below the skin of the mannequin. On the virtual reality side, the version 2.0 of *libptk* is fully functional and allows us to create and run various simulations such as controlling in real time (through wire less network links) a set of mobile robots or a complex scene involving deformable organs and surgical tools.

A first prototype of an endoscopic surgery training station was realized and showed the feasibility of such a simulator. This first prototype showed us the points to be improved and helped us to know the focus has to be put. A demonstration of a shared simulation among distant users was performed in Bern (CAOS Symposium).

For the haptic device, work has to be done on the design and implementation of efficient third and fourth degrees of freedom. First tests and prototypes have already been realized and are currently evaluated.

On the virtual reality emphasis is put on three topics and physical properties such as bleedings, fat tissues modelization and gravity integration.

Bleeding is an important feature to simulate to warn a trainee if she or he touches a soft organ (such as the liver) with a sharp tool. Just as we are using animated textures to get a pulsation-like aspect, we are using the same technique to simulate bleedings by covering parts of a solid with a dynamic texture. The same trick will also be used to simulate the *smoke* generated when the surgeon burns fat tissues. Tests showed that the results achieved so far are very promising.

An important step of the operation we are focusing on, is the *removal of fat tissues*. First test showed that considering fat tissues as deformable solids and using a deformation technique with no relaxation of the touched points is a very promising technique.

The force profile technique allows us to perform real time simple deformations, but to generate more anatomical-like behaviors, they need to be improved in a few directions. For instance, we need to take in account the gravity field to get differences on deformation depending on the touched area (top, bottom or sides) of the organ. We also need to define anchors, or fixed points to link organs to each other, so a deformation will be propagated from an organ to another according to their different physical properties.

5. Conclusion

After two years of work, the first realized prototype (including force feedback devices) showed that our approach can lead to an efficient and usable training station for endoscopic surgery. Since the goal is to provide surgeons with a realistic simulator, the strategy we use is to find tradeoffs between the realism to obtain and real time constraints. For instance, real time deformations could not have been performed by using finite elements methods, so we simplified the technique to get a real time behavior while asking surgeons if the system is realistic enough. Since the simulator is designed as an *agent*, users can share simulations through network links and therefore work together where ever they are located. Finally, the simulation to run is fully described in a file (read at initialization) where the operator describes the scene, so the same software can be used for various kinds of shared world applications.

Bibliography

☯1☸ Philip R. Cohen & Adam J. Cheyer, An Open Agent Architecture, AAAI Spring Symposium, 1994 (http://www.ai.sri.com/~oaa/)

☯2☸ R. Baumann, Force feedback for minimally minimal invasive surgery simulator, Medicine meets virtual reality 4, San Diego, CA, January 1996, pp 564-579.