Real-Time Smoke and Bleeding Simulation in Virtual Surgery

Release 1.00

Stefan Daenzer[1], Kevin Montgomery[2],
Roland Unterhinninghofen[3], Rüdiger Dillmann[4]

July 1, 2006

[1]Stanford University, stefan.daenzer@stanford.edu
[2]Stanford University, kevin@cohiba.stanford.edu
[3]Universität Karlsruhe, uhhofen@ira.uka.de
[4]Universität Karlsruhe, dillmann@ira.uka.de

Abstract

We describe the implementation of smoke and bleeding simulation in the Open Source Surgery Simulator SPRING [9], which is particularly targeted for minimally invasive surgery simulation. Many smoke and bleeding simulations offer high physical and visual accuracy, but the underlaying models are to complex to run in real-time while performing soft-tissue simulation, collision detection and haptic device support at the same time. Our algortihms are based on simple models, that allow the surgery simulation to run in real-time.

Contents

Visualizations of various physical and natural phenomena, such as smoke and fluids, have been widely used in computer graphics and simulation. Also in the surgical training field, surgery simulators play a great role, especially for minimally invasive surgery (laparoscopy). In contrast to open surgery, laparoscopic surgery is peformed by inserting camera and tools through small trocars into the patient. It has been widely proofed that laparoscopic surgery causes less tissue damage than Open Surgery, and that the recovery time for patients is shorter. Laparoscopic surgery is harder to perform though, as surgeons need more training to perform complex surgeries. The training effect of surgical simulators has been proofed by many studies [13] [4].

Training of laparoscopic surgical tasks with surgical simulators can be performed under similar conditions to real surgery, as the surgeon has a two dimensional display of the camera image. Haptically enabled tools can provide the virtual surgery with a realisic feeling. Visual effects, such as smoke and bleeding, enhance the quality of the simulation and enable a surgical simulator to simulate a great variety of surgical tasks, such as cauterizing and cutting. The quality of the visual effects is limited by pefromance issues. The main part of the computational power is needed for soft-tissue simulation, collision detection between objects and updating haptic devices.

Our smoke simulation is based on an uncoupled particle system. The particle system simulation is based on a very simple Newton solver. Our bleeding simulation is based on the underlying physical model of the bleeding object.

1 Related Work

There are many approaches to simulate the behavior of fluids. Grid based techniques, such as surface and heightfield methods were used by Kass [6] to simulate the surface of waves in water of varying depth. They used a heightfield to represent the fluid surface. Later Raghupathi [11] used a similar method to simulate the behavior of accumulated fluids in in real-time. Another grid based approach was done by Foster [5], including complex fluid behavior such as rotational eddies, spalshing and vorticity as well as interaction with rigid obstacles. This method solves the full Navier-Stokes equations in three dimensions on a grid with low resolution. Stam introduced a new technique to solve the Navier-Stokes equations more efficiently on a grid [14]. The method is extremely stable, regardless of the timestep between two iterations. Stam used his method to simulate smoke in real-time.

The big disadvantages of grid based techniques are the limitations in interaction with deformable and interactive models. If the environment changes, the grid has to be recomputed, what is very time consuming. Particle systems are another approach of fluid simulation. It was firstly introduced by Reeves [12]. Uncoupled particle systems, where no interaction between the particles is simulated, are used for smoke and blood-drop Simulation by Andersson [2]. Coupled particle systems, where the particles interact with each other, are used by Murta and Miller [1] to simulate splashing and dripping behavior of fluids. Smoothed particle hydrodynamics (SPH) is another particle based approach, where a continuous density field from a set of particles is computed. This desity field is then visualized, rather than visualizing every particle on its own. SPH was used by Mueller [7] [8] to simulate the interaction of water with solid bodies almost in real-time.

2 Environment

The simulation of smoke and bleeding has been implemented into the spen source surgical simulator SPRING [9]. SPRING is a real-time surgical simulation system with soft-tissue modeling and multi-user, multi-instrument networked haptics [10]. SPRING is cross-platform and runs on Unix, Windows and Linux. SPRING is written in C++, uses GLUT, GLUI and MUI for user interfaces and OpenGL for the visualization. Virtual tools can be defined in SPRING [3], capable of simulating various behaviors and interactions with other models. SPRING is available as an open soruce project at Sourceforge (`http://sourceforge.net/projects/spring-sim`).

3 Implementation

Our smoke simulation is based on an uncoupled particle system. No collision detection between the smoke and the surrounding objects in the virtual environment is performed, what makes updating the particles much faster. The smoke particles are then visualized as smoke puffs, by mapping a two dimensional texture on a rectangle at each particle's position. In our surgical simulator environment we can add virtual, haptically enabled tools. For the smoke simulation we created a new class of tools. With these tools we can simulate burning behavior of tools. The model of these tools consist of points in three dimensional space, edges defined by a set of two nodes and faces, defined by three nodes. A subset of these edges and faces can be defined as burning.

SPRING is capable of collision detection between objects, such as a soft tissue model and a tool. The collsions of two objects are stored within SPRING in a data structure called collision pairs. If there is a collision between two objects, SPRING creates a collision pair of which each holds a pointer to the colliding faces or edges of the objects involved. If one of the collision pairs holds a pointer to a edge or face that is defined as burning, the smoke simulation is initialized. A `particleGenerator` class is instantiated, releasing particles at the tip of the burning tool. The behavior of the soft-tissue simulation and the particle system are separated, the particle system is solved by the basic rules of behavior of masses under the influence of external forces (e.g. Newton). The `particleGenerator` object is parameterized with the maximum number of particles that are in the simulation simulataneously, the particle mass, the fade-in time, the maximum opacity of the particles, the time the particle fades in (e.g. the time till the particle reaches the maximum opacity), the time the particle fades out (e.g. the time till the particle reaches zero opacity after maximum opacity) the particle lifetime and the release velocity with which the particles are parameterized when released.

Also different kinds of forces are are attached to the particles, so the particles behave like masses under the influence of the attached forces. Different kinds of forces can be simulated by the particle system. For the smoke simulation, just gravitational forces and forces directed to a position in three dimensional space are used.

The bleeding visualization is capable of simulating blood trails on the surface of the bleeding model. There have been two different approaches imlemented. The first approach draws the blood trails as simple 2-D lines along the edges of the bleeding model. The second approach draws a three dimensional ellipse along the edges of the model. As for the smoke simulation, we created a new class of tools. With these tools we can simulate cutting tools. With the same approach as for the burning tools, we define a subset of the edges of the tool as sharp. If there is a collision between a sharp edge with another model's edge or face, the nearest node to that position is calculated and the pointer is stored. A pathfinding algorithm is then initialized, which searches for the edge connected to that node that has the highest partial derivate for y and lies on the surface of the model. The node connected to that edge is determined and the pointer is stored. This continues until the algorithm processes a node that has no node with a negative partial derivate of y connected by an edge to it. The calculated path is then visulized by one of the two bleeding visualization techniques.

4 Results

We set up a few scenes to show the capabilities of our simulation. Figure 1 shows the tip of a hot instrument touching the surface of an untextured model of a gallbladder. The smoke rises up from the contact point and fades out over time. In Figure 2, a sharp edge of a tool has stiched the surface of a textured gallbladder model. The blood trail is visualized as a two dimensional linestrip along the model. The blood follows the

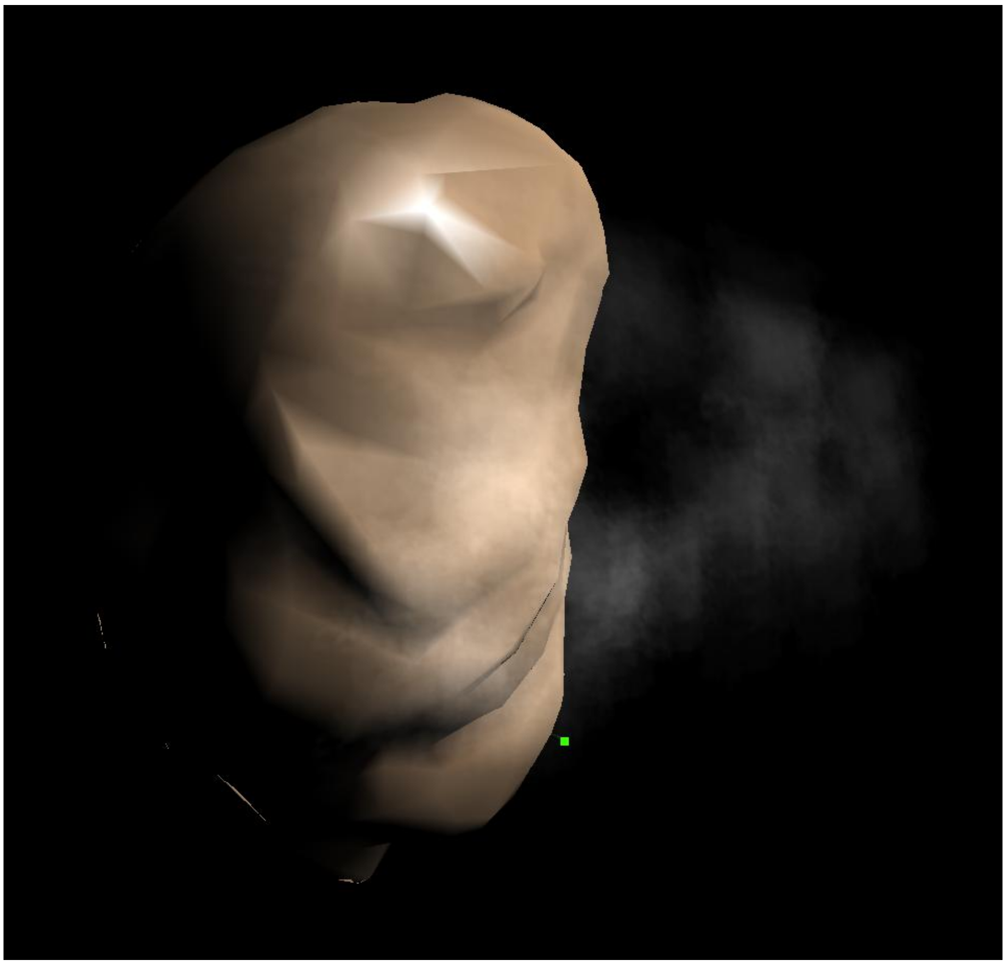

Figure 1: Smoke is rising up, as the hot instument touches the surface of the gallbladder

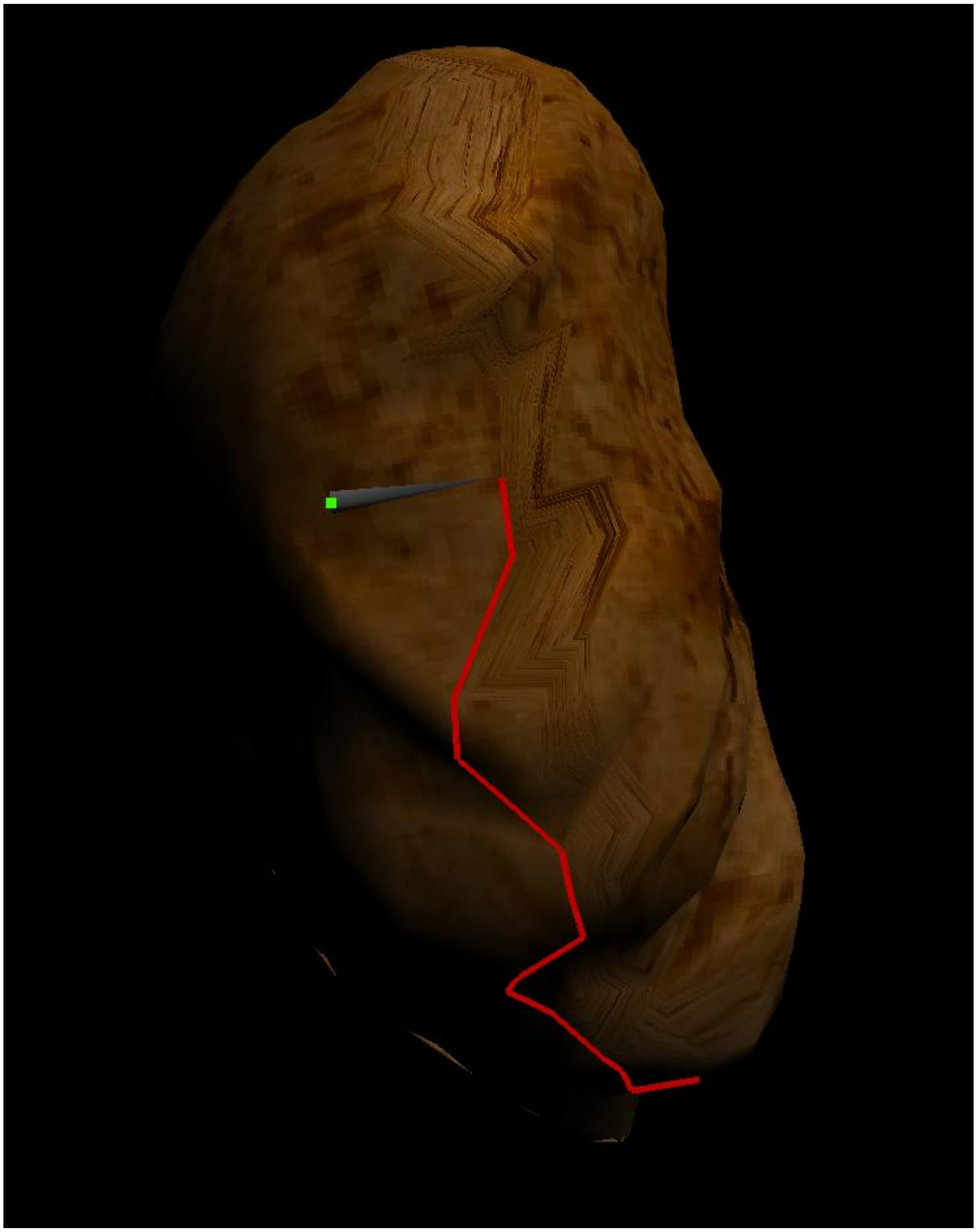

Figure 2: The blood trail visualized as a two dimensional line strip, flowing down on a model of the gallbladder

Figure 3: The blood trail visualized as a three dimensional ellipse strip, flowing down on a model of the gallbladder

shape of the model as it flows down. Figure 3 shows the blood visualized as a three dimensional ellipse strip on the model of the gallbladder.

We used the following libraries to develop SPRING and the particle system:

- OpenGl
- GLUT
- GLUI
- MUI
- GLAUX

All libraries listed above are delivered with the distribution of SPRING, that can be found at Sourceforge (`http://sourceforge.net/projects/spring-sim`).

References

[1] J. Miller A. Murta. Modelling and rendering liquids in motion. *Proceedings of WSCG*, 1999. 1

[2] Lars Andersson. Real-time fluid dynamics for virtual surgery. *Masters Thesis*, 2005. 1

[3] Kevin Montgomery Cynthia D. Bruyns. Generalized interactions using virtual tools within the spring framework: Probing, piercing, cauterizing and ablating. *Medicine Meets Virtual Reality*, 2002. 2

[4] M. Downes, M. Cavusoglu, W. Gantert, L. Way, and F. Tendick. Virtual environments for training critical skills in laparoscopic surgery, 1998. (document)

[5] Nick Foster and Dimitri Metaxas. Realistic animation of liquids. *Graphical models and image processing: GMIP*, 58(5):471–483, 1996. 1

[6] G. Miller M. Kass. Rapid, stable fluid dynamics for computer graphics. *SIGGRAPH Conference Precedings pp. 49-57*, 1990. 1

[7] M. Gross M. Mueller, D. Charypar. Particle-based fluid simulation for interactive applications. *Proceeding of 2003 ACM SIGGRAPH Symposium on Computer Animation*, 2003. 1

[8] M. Teschner M. Mueller, S. Schirm. Interactive blood simulation for virtual surgery based on smoothed particle hydrodynamics. *Journal of Technology and Health Care*, 2004. 1

[9] Kevin Montgomery. Spring: A general framework for collaborative, real-time surgical simulation. *Medicine Meets Virtual Reality*, 2002. (document), 2

[10] Kevin Montgomery. Project hydra- a new paradigm of internet-based surgical simulation. *Medicine Meets Virtual Reality*, 2006. 2

[11] L. Raghupathi. Simulation of bleeding and other visual effects for virtual laparoscopic surgery. *Master thesis*, 2002. 1

[12] W. T. Reeves. Particle systems - a technique for modeling a class of fuzzy objects. *Computer Graphics*, 17, 1983. 1

[13] Guy Sela, Sagi Schein, and Gershon Elber. Real-time incision simulation using discontinuous free form deformation. (document)

[14] J. Stam. Stable fluids. *SIGGRAPH*, 1999. 1

Real-Time Smoke and Bleeding Simulation in Virtual Surgery

Release 1.10

Stefan Daenzer[1], Kevin Montgomery[2],
Rüdiger Dillmann[3], Roland Unterhinninghofen[4]

July 20, 2006

[1]Stanford University, stefan.daenzer@stanford.edu
[2]Stanford University, kevin@cohiba.stanford.edu
[3]Universität Karlsruhe, dillmann@ira.uka.de
[4]Universität Karlsruhe, uhhofen@ira.uka.de

Abstract

We describe the implementation of smoke and bleeding simulation in the open source surgery simulator SPRING [9], which is particularly targeted for minimally invasive surgery simulation. Many smoke and bleeding simulations offer high physical and visual accuracy, but the underlaying models are to complex to run in real-time while performing soft-tissue simulation, collision detection and haptic device support at the same time. Our algorithms are based on simple models, that allow the surgery simulation to run in real-time.

Contents

Visualizations of various physical and natural phenomena, such as smoke and fluids, have been widely used in computer graphics and simulation. Also in the surgical training field, surgery simulators play a great role, especially for minimally invasive surgery (laparoscopy). In contrast to open surgery, laparoscopic surgery is performed by inserting camera and tools through small trocars into the patient. It has been widely proofed that laparoscopic surgery causes less tissue damage than Open Surgery, and that the recovery time

for patients is shorter. Laparoscopic surgery is harder to perform though, as surgeons need more training to perform complex surgeries. The training effect of surgical simulators has been proofed by many studies [13] [4].
Training of laparoscopic surgical tasks with surgical simulators can be performed under similar conditions to real surgery, as the surgeon has a two dimensional display of the camera image. Haptically enabled tools can provide the virtual surgery with a realistic feeling. Visual effects, such as smoke and bleeding, enhance the quality of the simulation and enable a surgical simulator to simulate a great variety of surgical tasks, such as cauterizing and cutting. The quality of the visual effects is limited by performance issues. The main part of the computational power is needed for soft-tissue simulation, collision detection between objects and updating haptic devices.
Our smoke simulation is based on an uncoupled particle system. The particle system simulation is based on a very simple Newton solver. Our bleeding simulation is based on the underlying physical model of the bleeding object.

1 Related Work

There are many approaches to simulate the behavior of fluids. Grid based techniques, such as surface and height field methods were used by Kass [6] to simulate the surface of waves in water of varying depth. They used a height field to represent the fluid surface. Later Raghupathi [11] used a similar method to simulate the behavior of accumulated fluids in in real-time. Another grid based approach was done by Foster [5], including complex fluid behavior such as rotational eddies, splashing and vorticity as well as interaction with rigid obstacles. This method solves the full Navier-Stokes equations in three dimensions on a grid with low resolution. Stam introduced a new technique to solve the Navier-Stokes equations more efficiently on a grid [14]. The method is extremely stable, regardless of the timestep between two iterations. Stam used his method to simulate smoke in real-time.
The big disadvantages of grid based techniques are the limitations in interaction with deformable and interactive models. If the environment changes, the grid has to be recomputed, what is very time consuming. Particle systems are another approach of fluid simulation. It was firstly introduced by Reeves [12]. Uncoupled particle systems, where no interaction between the particles is simulated, are used for smoke and blood-drop Simulation by Andersson [2]. Coupled particle systems, where the particles interact with each other, are used by Murta and Miller [1] to simulate splashing and dripping behavior of fluids. Smoothed particle hydrodynamics (SPH) is another particle based approach, where a continuous density field from a set of particles is computed. This density field is then visualized, rather than visualizing every particle on its own. SPH was used by Mueller [7] [8] to simulate the interaction of water with solid bodies almost in real-time.

2 Environment

The simulation of smoke and bleeding has been implemented into the open source surgical simulator SPRING [9]. SPRING is a real-time surgical simulation system with soft-tissue modeling and multi-user, multi-instrument networked haptics [10]. SPRING is cross-platform and runs on Unix, Windows and Linux. SPRING is written in C++, uses GLUT, GLUI and MUI for user interfaces and OpenGL for visualization. SPRING's soft-tissue simulation is is based on a mass-spring model. Virtual tools can be defined in SPRING [3], capable of simulating various behaviors and interactions with other models. SPRING is available as an open source project at Sourceforge (`http://sourceforge.net/projects/spring-sim`).

3 Implementation

Our smoke simulation is based on an uncoupled particle system. The behavior of the system follows the trajectories of N particles over any time interval. Given initial positions and velocities, the trajectory of each particle is governed by Newtons second law of motion,

$$m_i \frac{d^2 x_i}{dt_i^2} = -\nabla_i \Phi, i \in \{1, ..., N\} \tag{1}$$

where m_i ist the mass, and x_i is the position, of the ith particle. The force is determined from the gradient of the potential function Φ. No collision detection between the smoke and the surrounding objects in the virtual environment is performed, what makes updating the particles much faster. The smoke particles are then visualized using a fast spherical billboarding enabled two-dimensional texture mapping technique. For the smoke simulation we created a new class of tools. With these tools we can simulate burning behavior by defining subsets of their edges and faces as hot. The type and behavior of the generated smoke can be varied by the parameters of the particle system and the texture mapping. So the smoke simulation could be used in a wide range of applications.

The bleeding visualization is capable of simulating blood trails on the surface of the bleeding model. We introduce a fast model based approach. The blood is bound to the nodes that define the model of the geometric object, the blood is flowing on. By binding the flow to the nodes, no collision detection between the blood and the model has to be done. As for the smoke simulation, we created a new class of tools. With these tools we can simulate cutting tools. With the same approach as for the burning tools, we define a subset of the edges of the tool as sharp. For the visualization of these blood-trails, three different visualization techniques have been implemented. The first visualization draws the blood trails as simple two dimensional lines along the edges of the bleeding model. The second technique draws the blood trail as textured Lines along the model. The third visualization draws a three dimensional ellipse along the edges of the model.

4 Results

We set up a few scenes to show the capabilities of our simulation. In figure 1 smoke arises as a hot instrument touches the surface of the model. Figure **??** shows the blood trails visualized as two dimensional polygon strips, after being cut by a sharp instrument. Figure 3 is a visualization of the blood trail visualized as a textured polygon strip. Figure **??** shows the three dimensional ellipse strip visualization.

We used the following libraries to develop SPRING and the particle system:

- OpenGl
- GLUT
- GLUI
- MUI
- GLAUX

All libraries listed above are delivered with the distribution of SPRING, that can be found at Sourceforge (http://sourceforge.net/projects/spring-sim).

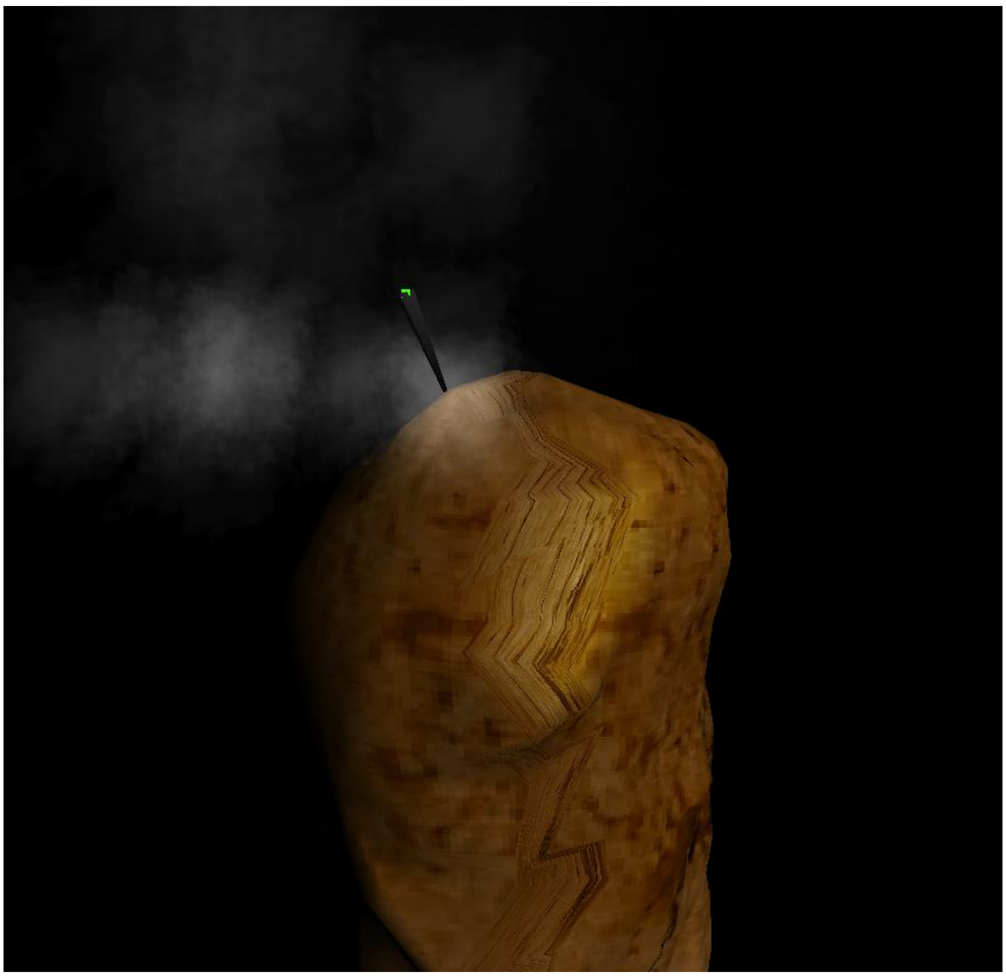

Figure 1: Smoke is rising up, as the hot instrument touches the surface of the gallbladder

Figure 2: The blood trail visualized as colored polygons, flowing down on a model of the gallbladder

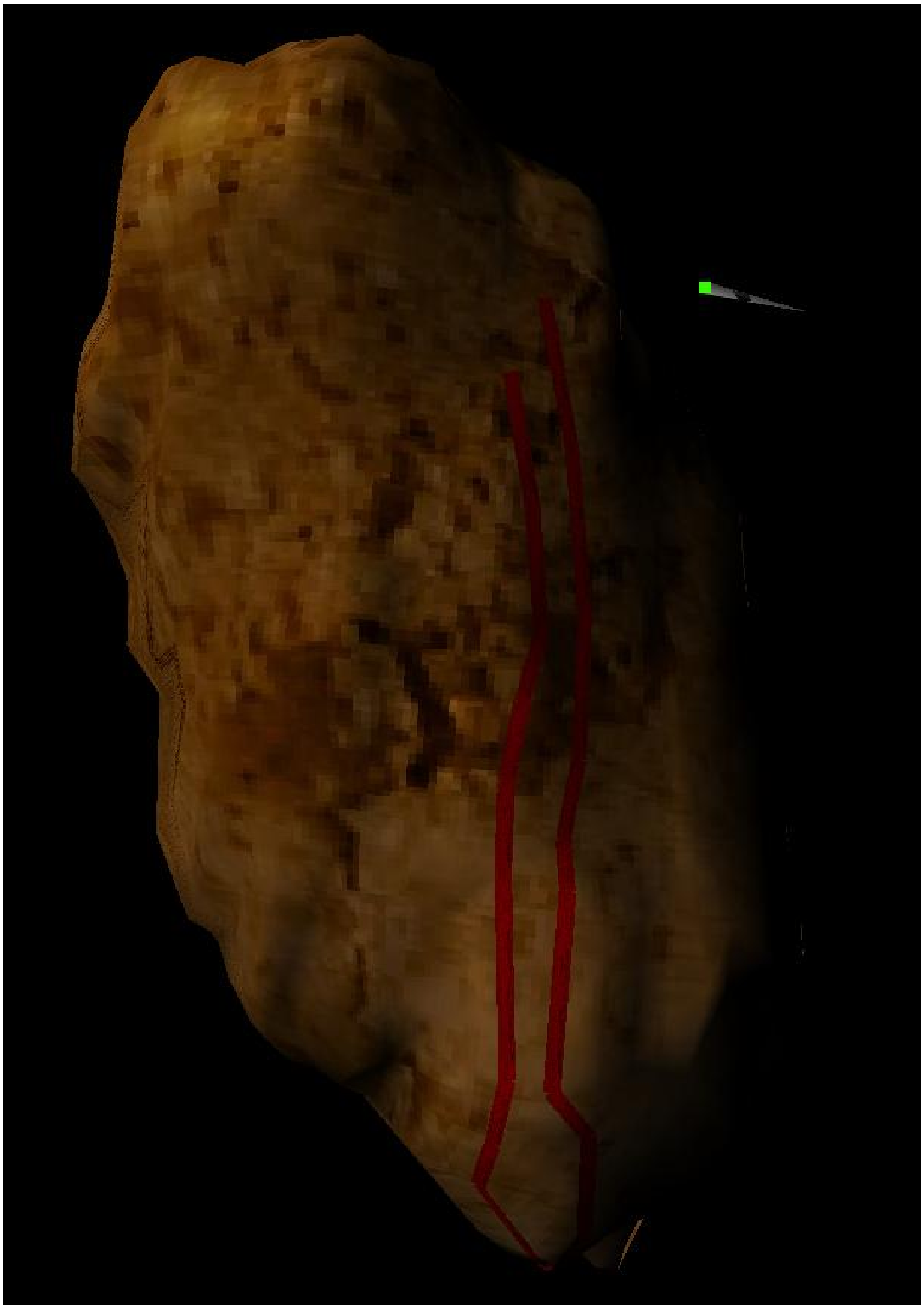

Figure 3: The blood trail visualized as textured polygons, flowing down on a model of a gallbladder

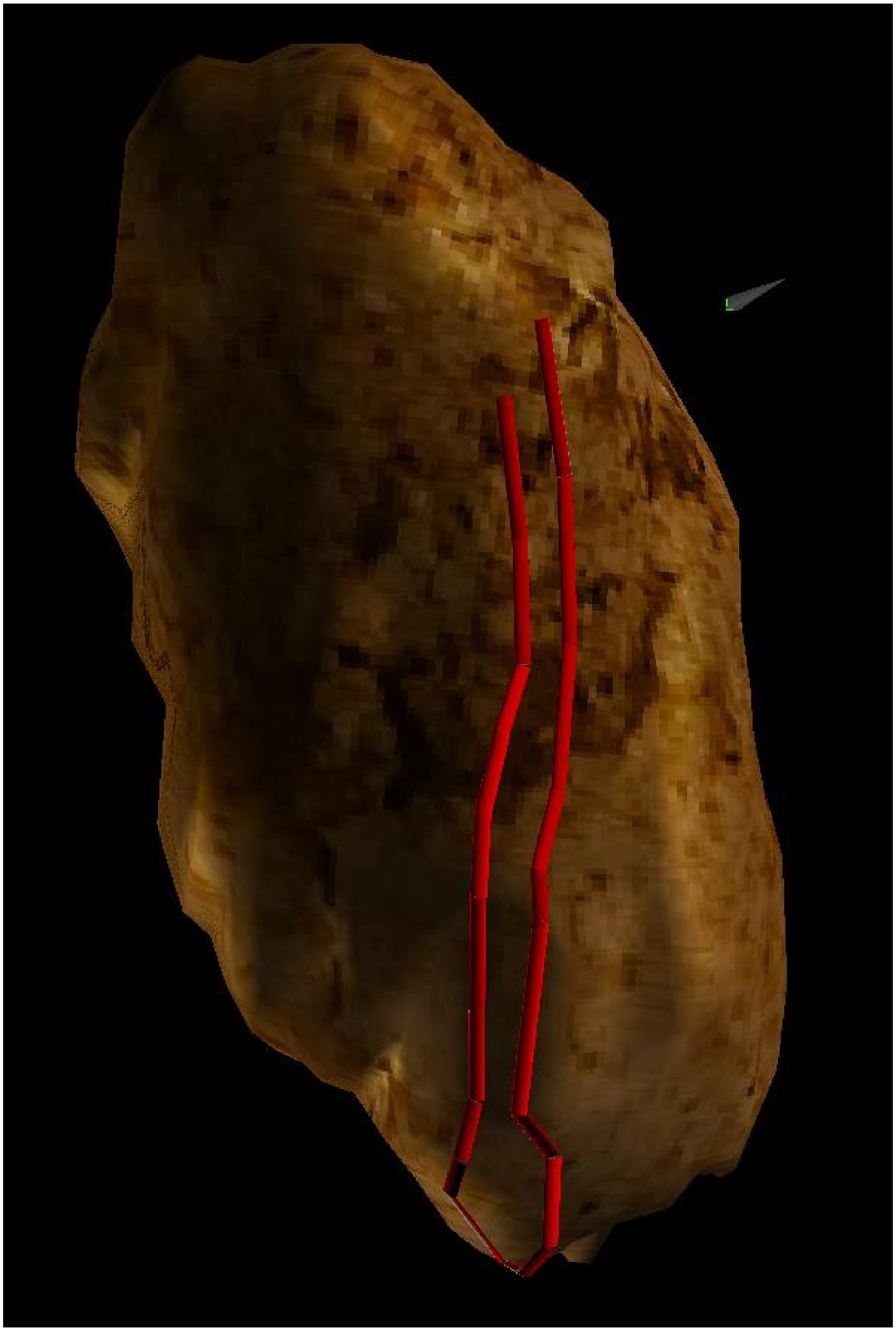

Figure 4: The blood trail visualized as a three dimensional ellipse strip, flowing down on a model of the gallbladder

References

[1] J. Miller A. Murta. Modelling and rendering liquids in motion. *Proceedings of WSCG*, 1999. 1

[2] Lars Andersson. Real-time fluid dynamics for virtual surgery. *Masters Thesis*, 2005. 1

[3] Kevin Montgomery Cynthia D. Bruyns. Generalized interactions using virtual tools within the spring framework: Probing, piercing, cauterizing and ablating. *Medicine Meets Virtual Reality*, 2002. 2

[4] M. Downes, M. Cavusoglu, W. Gantert, L. Way, and F. Tendick. Virtual environments for training critical skills in laparoscopic surgery, 1998. (document)

[5] Nick Foster and Dimitri Metaxas. Realistic animation of liquids. *Graphical models and image processing: GMIP*, 58(5):471–483, 1996. 1

[6] G. Miller M. Kass. Rapid, stable fluid dynamics for computer graphics. *SIGGRAPH Conference Precedings pp. 49-57*, 1990. 1

[7] M. Gross M. Mueller, D. Charypar. Particle-based fluid simulation for interactive applications. *Proceeding of 2003 ACM SIGGRAPH Symposium on Computer Animation*, 2003. 1

[8] M. Teschner M. Mueller, S. Schirm. Interactive blood simulation for virtual surgery based on smoothed particle hydrodynamics. *Journal of Technology and Health Care*, 2004. 1

[9] Kevin Montgomery. Spring: A general framework for collaborative, real-time surgical simulation. *Medicine Meets Virtual Reality*, 2002. (document), 2

[10] Kevin Montgomery. Project hydra- a new paradigm of internet-based surgical simulation. *Medicine Meets Virtual Reality*, 2006. 2

[11] L. Raghupathi. Simulation of bleeding and other visual effects for virtual laparoscopic surgery. *Master thesis*, 2002. 1

[12] W. T. Reeves. Particle systems - a technique for modeling a class of fuzzy objects. *Computer Graphics*, 17, 1983. 1

[13] Guy Sela, Sagi Schein, and Gershon Elber. Real-time incision simulation using discontinuous free form deformation. (document)

[14] J. Stam. Stable fluids. *SIGGRAPH*, 1999. 1