Pisa takes a stand for responsibility in healthcare and medical technology
6th Annual HCTM Conference -HOF- Scuola Superiore Sant'Anna
3-5 October 2007, Pisa, Italy

New techniques for computer-based simulation in surgical training

Giuseppe Turini*

EndoCAS Center, Università di Pisa
Ospedale di Cisanello, Via Paradisa 2, I-56124, Pisa, Italy
E-mail: giuseppe.turini@crim.sssup.it
*Corresponding author

Nico Pietroni

Visual Computing Lab
ISTI Istituto di Scienza e Tecnologie dell'Informazione
Area della Ricerca CNR, Via G. Moruzzi 1, I-56124, Pisa, Italy
E-mail: nico.pietroni@isti.cnr.it

Giuseppe Megali

CRIM, Polo Sant'Anna Valdera, Scuola Superiore Sant'Anna
Viale Rinaldo Piaggio 34, I-56025, Pontedera (Pisa), Italy
E-mail: peppe@crim.sssup.it

Andrea Pietrabissa

Centro EndoCAS, Dipartimento di Oncologia, dei Trapianti e delle Nuove Tecnologie in Medicina, Università di Pisa
Ospedale di Cisanello, Via Paradisa 2, I-56124, Pisa, Italy
E-mail: a.pietrab@med.unipi.it

Abstract: In the recent decades robotics and computer science have been gaining more and more relevance in all aspects of our lives. In surgery, for example, they gave birth to procedures, impossible to perform otherwise, like the tele-surgery or the nano-surgery. On this regard, these applied sciences already play an important role in assisting the surgeon both in the operative room and, as a support, in the education of young surgeons, but much work has still to be done.

In fact in these last years we have seen an extreme change in the traditional training in surgery and the computer-based simulation is one of the main reason of this shift. The spread of Minimally Invasive Surgery (MIS) has brought major improvements in the quality of healthcare, but it has also increased the complexity of the surgical procedures requiring advanced and highly specialized training systems. Moreover these training procedures need to be reiterated during the operational life of surgeons. Therefore, considering the limited availability of cadavers and the public concern with the non-ethical treatment of animals, the traditional approaches to surgical training are drastically limited encouraging the use of surgical simulators based on virtual environments.

Healthcare industries and the scientific community in medicine agree indicating the disruptive potential of the application of Virtual Reality (VR) to the training in the medical field. Therefore the next step is the development of surgical simulators with an high level of realism in order to practice complex procedures in a safe environment. Moreover it is decisive that this evolution is done integrating advanced medical imaging and processing, allowing surgeons to practice simulated interventions on patient specific dataset.

The increasing importance of MIS techniques will cause a drastic change in pre-operation planning and basic surgical training. In fact, the features of this kind of surgical approach (the workspace limitation, the 2D vision through a laparoscopic camera and the indirect physical interaction with the patient body) make it possible to use a surgical simulator to train, plan or simulate an intervention, reproducing the visual and tactile feedback of the real surgical procedure on a real patient.

This paper presents some research and applicative results on Computer Assisted Surgery (CAS) achieved in the framework of EndoCAS, a newly founded Center of Excellence in Pisa. The research has involved: the development of segmentation algorithms for volumetric datasets, the simulation of bone drilling procedures, the modeling of deformable object cuts and deformations and the simulation of rope interactions during a suture procedure in MIS. All these projects were been developed using a new open source library to support the implementation of techniques for simulating deformable objects.

Our purpose is to enhance the surgical training with new improved techniques applied both to the medical imaging and to the computer-based simulation in order to carry the surgical training to a next level of realism.

Keywords: surgical simulation; computer aided surgery.

Biographical notes: Giuseppe Turini was born in 1975 in Pescia (Pistoia, Italy). He received the master degree in Computer Science from the University of Pisa in 2004, and in the same year he joined to the Visual Computing Lab of the CNR-ISTI in Pisa. Since 2005 he is a research fellow at the EndoCAS Centre, and his main research interests are in the field of computer graphics visualization and physical simulation. At the present time his research is carried out in the context of computer-assisted surgery.

Nico Pietroni was born in 1978 in Siena (Italy). He received the Laurea degree in Computer Science from the University of Pisa in 2004, and in the same year he joined to the Visual Computing Lab of the National Research Council in Pisa. He start his Ph.D. activity in 2005. His main research interests are in the field of Computer graphics, deformable models simulation and interaction.

Giuseppe Megali was born in 1972 in Reggio Calabria (Italy). He received the Laurea degree in Computer Science from the University of Pisa in 1998, and in the same year he joined to the MiTech Lab (now CRIM Lab) of the Scuola Superiore Sant'Anna in Pisa. He received his Ph.D. degree in Robotics from the University of Genova in 2002. From 2002 to 2004 he had a Post-Doctoral fellowship in Bioengineering at the Scuola Superiore Sant'Anna in Pisa. Since 2004 he is assistant professor of Biomedical Robotics at the Scuola Superiore Sant'Anna and coordinator of EndoCAS. His main research interests are in the field of biomedical robotics, computer-assisted surgery and analysis of surgical gesture. His research is carried out in the context of National and European projects in the field of biomedical engineering.

Andrea Pietrabissa was born in 1959 in Pisa. He graduated with honors in Medicine and Surgery at the University of Pisa and he specialized in General

Surgery. From 1986 to 1988 he was Research Fellow in Hepatic Transplantation at the University of Chicago, and from the 1992 to the 1993 he was Senior Registrar at the Ninewells Hospital in Dundee (Scotland) and Lecturer in Surgery at the University of Dundee. Since 1999 he is Associate Professor of General Surgery at the University of Pisa. He is responsible of the Minimally Invasive Section of the Department of General Surgery and Transplantation at the University of Pisa, and, until now, he performed over than 3000 surgical interventions. He is member of several scientific societies (EAES European Association of Endoscopic Surgeons, SAGES Society of American Gastrointestinal Surgeons), and he is in the international Editorial Board of the Surgical Endoscopy journal. He was involved in a lot of research projects at national and international level and co-inventor of some patents.

1 Introduction

The quality of life is strictly related to the progress in various fields of applied sciences. Usually these advancements are carried on by researchers with the same skills and in these situations the results are directly available. Using applied sciences in medicine is different, because there is a wide difference between the background of the developers of technology and the background of the surgeons, so the researchers can hardly communicate their needs correctly. Therefore to develop a simulator for non-invasive surgery we need the guide of trained surgeons as well as to develop a flight simulator we need the help of trained pilots.

EndoCAS is a Center for Endoscopic Computer Assisted Surgery located inside the Ospedale di Cisanello, in Pisa. The center was born to put together researchers from different scientific backgrounds that would not meet otherwise. It is the result of a joint proposal of the University of Pisa, of Scuola Superiore Sant'Anna (CRIM Lab) and the Visual Computing Laboratory (ISTI-CNR) in the framework of MIUR funding for centers of excellence. Its goals are to address key knowledge, technology, and systems design barriers that must be overcome in the development of CAS systems.

In this paper we overview some of the results obtained in the first years of activity of the center. In Section 2 we show a novel scheme for extracting anatomical organs from noisy Computed Tomography (CT) data based on active contours and marching cubes techniques. This approach was successfully used to reconstruct the vena cava from CT data where standard methods failed. In Section 3 a technique for simulating bone drilling is shown. The approach is very general and it can be adopted by all the approaches using tetrahedral meshes for representing the solid, with limited influence on the performance of the rest of the simulation. In Section 4 we briefly describe a fast and robust technique to perform interactive virtual cutting on deformable objects. Our algorithm is highly general and does not use any assumption neither on how the deformation function is computed nor on how the boundary of the object is encoded. Then in Section 5 we present a physically based model for real-time simulation of thread dynamics. Our approach leads to a stable real time simulation. Section 6 is dedicated to an open source library called IdoLib, created to develop simulation-oriented applications. Conclusions are exposed in Section 7 that completes the paper.

2 Segmentation of noisy data

The reconstruction of the external surface of an organ is an important task for 3D visualization of medical datasets and it can also be used to support quantitative diagnostic measurements. However, this assignment may be challenging even in the seemingly easy cases of visually uniform regions. In fact segmentation methods should provide a closed smooth surface avoiding leakages but not missing bifurcations with smaller vascularizations. For these reasons these algorithms need topology control, robustness against noise and an appropriate smoothing method for the surface generation in order to preserve small structures or local pathologies that characterize the organ.

Many surface reconstruction methods can be found in literature, each with specific advantages and drawbacks. Simple isosurface extraction (Lorensen and Cline, 1987) does not guarantee topology control and it is quite sensitive to noise. Holes and leakages are also a relevant problem for more advanced level sets or geodesic surfaces methods (Mallardi and Sethian, 1995; Caselles and Kimmel, 1995). To prevent irregularities, holes and leakages the best approach is to use a parametric deformable surface instead of an implicit curve, with the same approach as the classic balloon snakes (Cohen and Cohen, 1993). These methods (see McInerney and Terzopoulos, 1996; for a review of the first medical applications) are efficient and robust. The elastic forces avoid the surface block near a noisy pixel and prevent leakages.

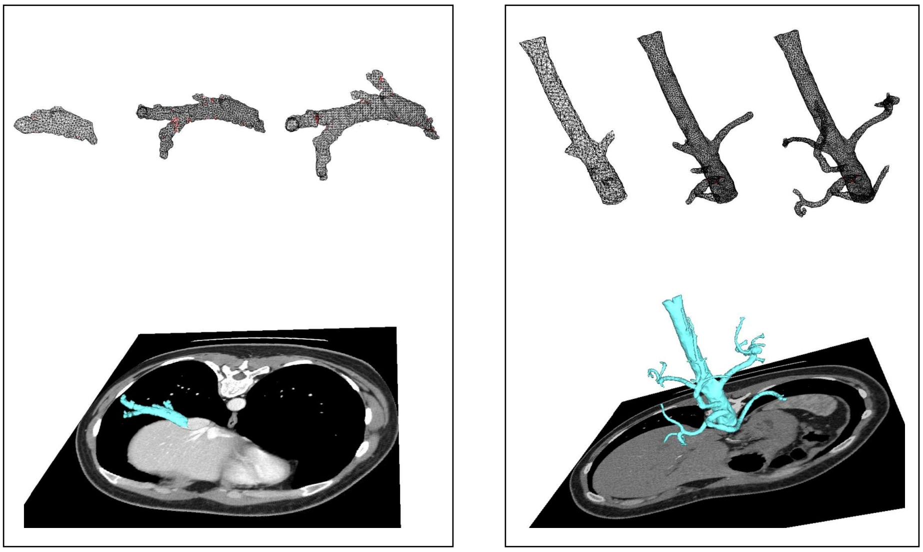

Figure 1 Different phases of extraction of vena cava (left) and aorta (right).

2.1 The balloon algorithm

The key idea is to place a very small deformable bubble inside the structure to be segmented and then inflate it. We control the inflation force using the density values stored in the voxels of the dataset, in this way we can drive automatically the inflation to fit the borders of the structure to be segmented. As the bubble grows, the triangles become bigger, but since the final goal of the process is to fit the bubble over the surface of the structure, it is self evident that the size of the triangles must be of the order of the size of the voxel. As a triangle grows more than a fixed threshold, we perform a simple mesh refinement, splitting the longest edge and adding two new triangles.

2.2 Self intersection management

During the inflation, the bubble mesh can intersect itself, because the anatomical structure to be segmented is not always topologically equivalent to a sphere. The detection of the intersections is needed to stop the bubble inflation. Therefore, to restore a non self-intersecting surface our algorithm performs a periodical collision detection in order to find intersecting portions of the surface and blocking the vertices belonging to intersecting faces. Then we build a distance map assigning to each voxel of the volume the signed distance from the closest non intersected face, which is negative if the point is inside the volume enclosed by the surface and positive otherwise. Due to the use of the part of the surface that is not self-intersecting , the result is that the isosurface of value 0 of the map is a surface defining the limits of the region of our interest with no self-intersections and the correct topology.

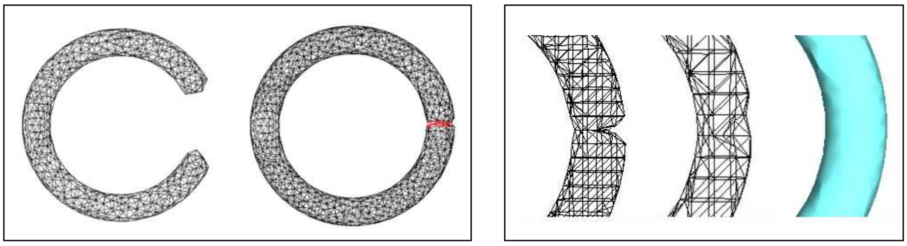

Figure 2 Example of self intersection management: the bubble expansion generate a self intersection (in red on the left) that is managed obtaining the final connected mesh (in cyan on the right).

2.3 Results

We have tested our algorithm on synthetic images of structures to validate the self collision detection/marching cubes method (see Figure 2). In all the tests, we have obtained a closed smooth surface defining an internal region close to the voxelized volume of interest.

The algorithm was applied to CT scans of the abdomen in order to segment the aorta and the vena cava. Figure 1 shows the results of these test cases. The segmentation is accurate and fast and no manual corrections are required.

3 Towards surgical simulator: bone drilling

The bone drilling procedure is an important task in many surgical interventions as: mastoidectomy, cochlear implantation or orbital surgery. The procedure mainly consists in the removal of part of the bone in contact with the surgical drill when the surgeon exerts sufficient pressure. The fact that the bone is an almost rigid material make the virtual bone object suitable for a regular discretization of the space, and therefore most of the approaches proposed in literature use voxel-based techniques that allow an easy bone material removal playing with the density in the voxels. Some examples of these techniques are: the *voxmap-pointshell* methods (McNeely and Puterbaugh, 1999; Petersik and Pflesser, 2002; Morris and Sewell, 2004), or the Graphics Processing Unit (GPU) *voxel-based techniques* (Agus and Giachetti, 2002). Mesh based approaches are also widely used to perform cuts (Bielser and Maiwald, 1999; Ganovelli and Cignoni, 2000; Niehuys and van der Stappen, 2000) and to render material removal (Agus and Gobbetti, 2006; Cotin and Ayache, 1999) but their accuracy depends on the size of the mesh elements. However, there are cases in which the bone drilling is only a part of the surgical procedure, and the same bone object can also be cut or, worse, slightly deformed and therefore voxel-based techniques do not work well.

We have developed a novel drilling method to overtake these drawbacks (Figure 3 left). Our algorithm is based on objects represented explicitly by means of tetrahedral meshes that can be easily generated from CT (or Magnetic Resonance) dataset. These volumetric meshes can be used to perform physical simulation of solids, they can be easily visualized and can also be used to simulate material with non uniform density. Unfortunately, since the number of mesh elements (tetrahedra) strongly influences the performance of the physical simulation, we have to face two opposite constraints: a high number of tetrahedra to achieve a realistic visualization of drilling and a low number of elements to execute the physical simulation in real-time.

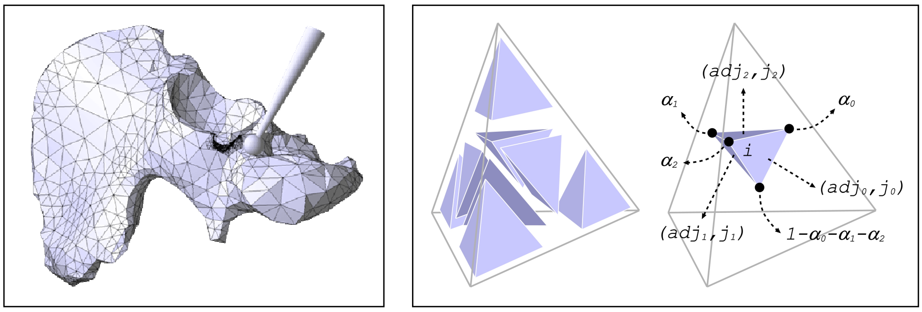

Figure 3 An example of drilling simulation (left) and the tetrahedral subdivision scheme (right).

3.1 Our drilling approach

Develop a drilling method that is independent from the physical simulation, means to decouple the drilling from the physical system. To do this we have decided to represent the drilling at sub-tetrahedral level defining a hierarchical tetrahedral decomposition scheme recursively splitting tetrahedra in 8 new tetrahedra (see Figure 3 right).

Moreover the algorithm manages the collisions between the tool and the bone mesh, so when a tetrahedron come in contact with the tool, the decomposition scheme is applied to adapt the mesh detail to the tool shape, in this way we can easily perform the bone material removal simply eliminating intersecting tetrahedra from the hierarchy. Hence the subtraction of a region of a tetrahedron can be represented computing a frontier in the hierarchy and tagging the nodes of the hierarchy inside the drilled region as *erased* (see Figure 4 top).

To optimize the refinement step during the simulation we have developed a pre-processing step to compute some of the data needed to apply the decomposition scheme. Then these data are stored in a static *Tree Table* to ensure fast direct accesses. Moreover we have used another data structure to implement a spatial indexing technique essential to speed up the collision detection. Then a list of visible node faces *v_list* was added to each tetrahedron to reduce the rendering time.

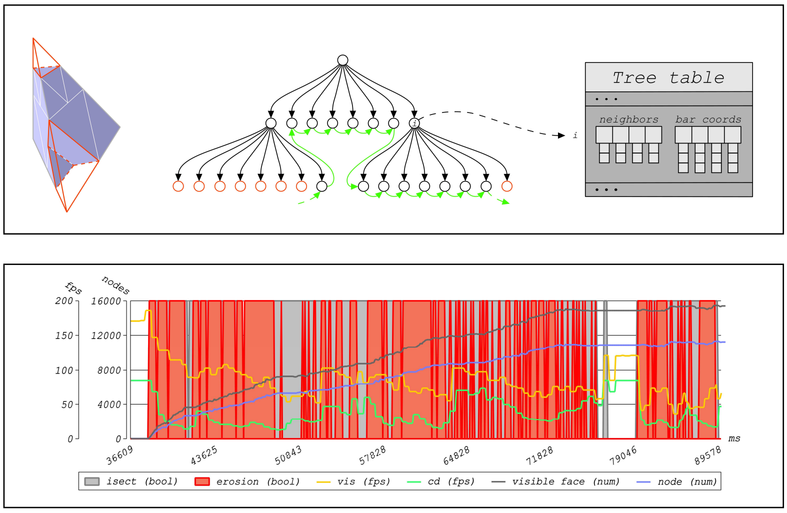

Figure 4 Data structures (top): a tetrahedron eroded (top-left), its hierarchy (top-middle) and the associated entry in the *TreeTable* (top-right).
Results (bottom): data produced during a basic test.

Our approach ensures two main advantages: first of all it is independent on the physical simulation technique and secondly the collision detection and the renderings (haptic and visual) are modeled at sub-tetrahedral level ensuring a high detail level.

3.2 The simulation system

The simulation system is a *multi-threaded* application formed by: the visual rendering thread, the collision detection thread and the haptic thread. The rendering thread simply renders all the visible faces of tetrahedra. If a tetrahedron has been eroded, then we run through its *v_list* and render each visible face of the nodes in the list. The collision handling thread manages the collision detection using a spatial hashing technique (Teschner and Heidelberger, 2003). This phase is crucial to determine all the actions to be performed on the tetrahedra (*Refine* and *Erase*) and the feedback to return to the haptic interface. The haptic interaction thread updates the tool position accordingly to the haptic interface movements and ensure high rendering frequency of force feedback that is essential to simulate a realistic haptic interaction.

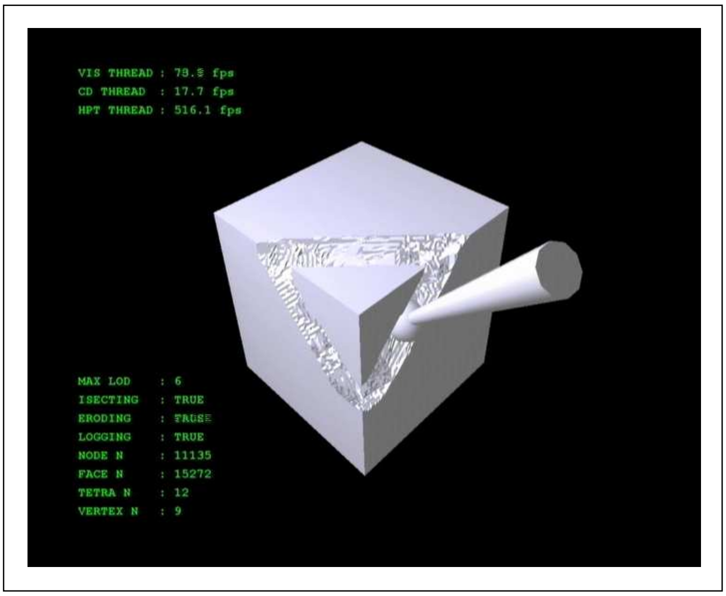

Figure 5 A screenshot during the basic drilling test.

3.3 Results

We have tested the collision detection, the handling of geometry and the updating rate of the three threads. In the tests the object is still static, even if this is not assumed anywhere in the implementation. Figure 4 bottom shows the performance of the algorithm in a test case while Figure 5 shows a snapshot of the cube at the end of the process.

4 Deformable object modelling: a robust cutting technique

One of the most important task in surgical simulation is virtual cutting of deformable objects. Most of the methods that can be found in literature, adopt mesh-based techniques that allow dynamic modifications to animate topological changes (Ganovelli and Cignoni, 2000). Unfortunately the main drawback of these approaches is the fragmentation. Moreover, if the mesh is also used as a partition of the object in finite elements for numerical simulation, the quality of re-meshing directly influences the stability of the computation.

In the last years *mesh-free* methods (*MMs*) (Müller and Keiser, 2004; Pauly and Keiser, 2005; Steinemann and Otaduy, 2006), often used in fluid simulation, have been adopted in computer graphics to model deformable solids. The MMs approximate physical quantities using freely sampled masses. Moreover they avoid re-meshing providing the continuity of the physical quantities. On the other hand MMs also pose the problems regarding the representation of the surface and the modifications needed to represent discontinuities. Although the issue of modeling cuts in MMs has been faced in the context of applied mechanics, a solution to manage real-time virtual cutting still missing.

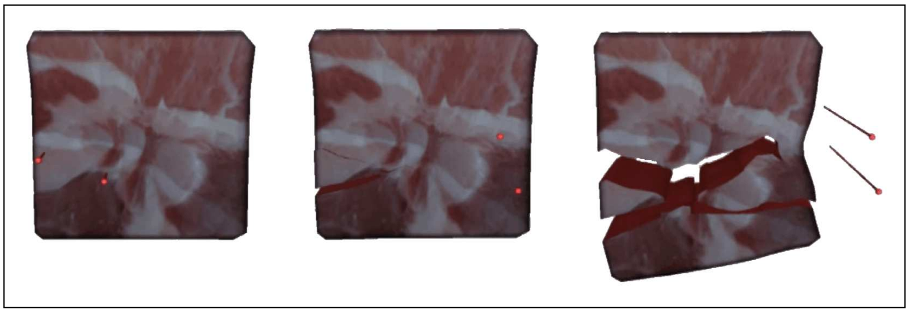

Figure 6 Few screenshots from a test example showing 2 rotating blades that cuts a solid cube of flesh.

4.1 The splitting-cubes algorithm

The key idea of our method is to embed the object in a regular grid and to define the tessellation of its surface on the base of the intersection of its boundary with the edges of the grid. As a consequence, when a cutting tool intersects the edges of the grid, the tessellation is implicitly adapted to show the new surface created by the cut.
The Splitting Cubes algorithm is highly general and does not use any assumption neither on how the deformation function is computed nor on how the boundary of the object is encoded, as long as its intersection with the edges of the grid can be computed.
We applied the Splitting Cubes to the case of the mesh-free method proposed in (Müller and Keiser, 2004), also introducing the Extended Visibility criterion to modify the shape functions to model the desired discontinuity of the deformation.

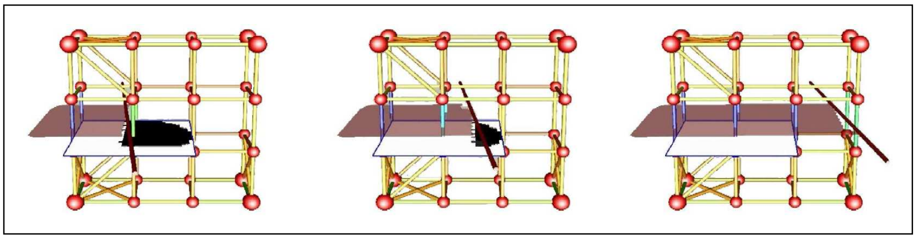

Figure 7 Three steps during a cut show how the *Extended Visibility Criterion* is used to update the phyxel connectivity.

4.2 The extended visibility criterion

Once the object is being cut and deformed we need to change the deformation function to reflect the discontinuity introduced. This mainly means to check the phyxel connectivity in an appropriate manner. Our approach relies on a visibility criterion that ensures smooth cuts and that can be implemented on the GPU optimizing the performance of the simulation.

4.2 Results

We have introduced the *Splitting Cubes*, a new algorithm for dynamic tesselation of an implicit surface interactively generated in response to cuts (Figure 6). Moreover we have also implemented an *Extended Visibility Criterion* (Figure 7), a novel GPU friendly solution to handle discontinuities in MMs taking advantage of the graphics hardware.

It is important to evidence that the Splitting Cubes algorithm only relies on a generic deformation function and on a description of the object's surface.

5 Robust interactive rope simulation

The main use of a thread simulator is certainly in endoscopic surgical simulation. Handling the surgical thread to make knots is one of the most difficult tasks for a surgeon because it requires ambidexterity with the endoscopic forceps.

Although the rope simulation can seem simpler than the simulation of more complex 3D structures, the interaction with a thread involves self-collision detection and contact handling. Furthermore, the thread is almost inextensible, and from the point of view of the simulation, this means that the methods modeling elasticity and using explicit time-integration schemes are poorly conditioned.

Most of the approaches in literature are energy-based (Wang and Burdet, 2006) and model the rope as a 1-dimensional chain of mass points connected by springs and physically simulated. However there are also non energy-based models (Brown and Latombe, 2004) that allow the making of complex knots in real-time even if the perception of the lack of physical simulation is clear.
Our method for simulating surgical thread is based on Position Based Dynamics (Müller and Heidelberger, 2007), the collision detection is carried out by spatial hashing.

5.1 Our approach

We simulate rope dynamic by defining a set of physical constraints. The simulation involves three steps: (1) moving the mass points according to their velocity and external action, (2) moving the mass points to satisfy the constraints and then (3) performing time integration. Position Based Dynamics is particularly useful to handle collision and contact, while it is hard to achieve stability using energy based methods, and this is the main reason for our choice.

The collision detection for the rope simulation is a crucial process. For this reason we have chosen to use a regular partition of the space cause the thread is divided in segments equally sized, so we have used a *Spatial Hashing* technique (Teschner and Heidelberger, 2003) with temporal marks.

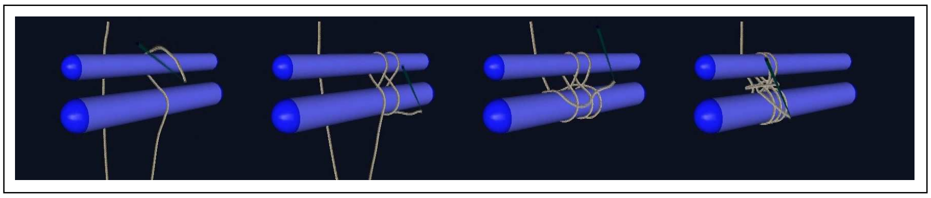

Figure 8 Four screenshots during the making of a complex knot around two tubes.

5.2 Results

Our test has shown that the model was efficient for the interactive thread simulation (Figure 8). The physically-based model for rope dynamics produces more realistic effects maintaining the stability and the controllability of the original Position Based Dynamics method.

6 IdoLib: Interactive Deformable Objects Library

IdoLib (http://idolib.sourceforge.net) is an open source C++ library initially developed at the Visual Computing Laboratory and now also developed at EndoCAS. The goal of IdoLib is to provide a set of basic tools and definitions for supporting the developer of a simulation-related application.

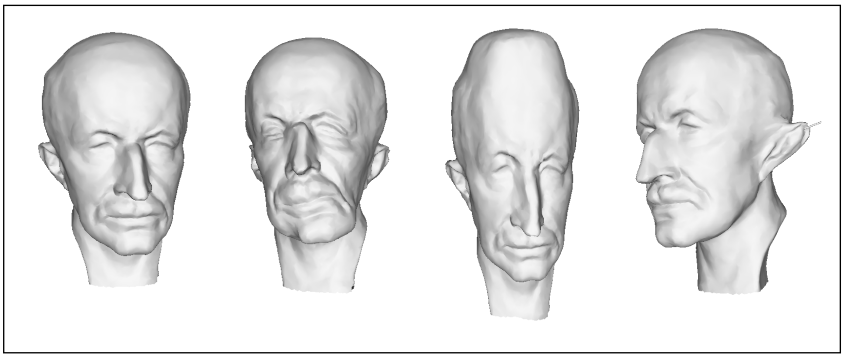

Figure 9 Deforming a mesh using IdoLib (MMs).

6.1 The core

IdoLib is specialized for explicit representation of the objects, both of the surface (triangular meshes, point based) and of the solid (tetrahedral meshes, point sampled volumes).

The core of IdoLib is based on two main identities: particles and relations. The particle is intended as a one dimension sampling of the volume with a parametric set of geometric and physical properties (position, velocity, acceleration, mass etc.) and the relation is, as the name says, a relation among a set of particles. For example in the

simplest case of the mass spring system the particle is the mass and the relation is the spring. The relation does not have to correspond to a geometric figure as for spring or triangles of tetrahedra: for example in the implementation of a point based method, the relation binds together all the particles closer than a given radius to each other. Figure 9 and Figure 10 show few examples of an application of the methods described above.
Moreover IdoLib provides a number of common integration schemes and a callback-based manner to add new schemes, so the user can switch among different integration schemes during the simulation. Similarly, the library is extendible and new methods can be easily added to those already implemented in IdoLib (refer to: Cotin and Ayache, 1999; O'Brien and Hodgins, 1999; Müller and Heidelberger, 2005; Costa and Balaniuk, 2001; Müller and Keiser, 2004; for the details). Note that IdoLib does not take control of the simulation loop and the main call only performs a single step of simulation.

Dependencies: IdoLib is written in C++ with STL library and relies only on another library, called VCG (Visualization and Computer Graphics Library, http://vcg.sourceforge.net), for the basic geometry structures and linear algebra.

6.2 Further work

IdoLib is still at a early stage. Although the structure is quite stable, further work is needed to provide a simple way to visualize the state of the simulation which is always essential in the debugging phase. Similarly, a complete documentation and examples is still to be done.

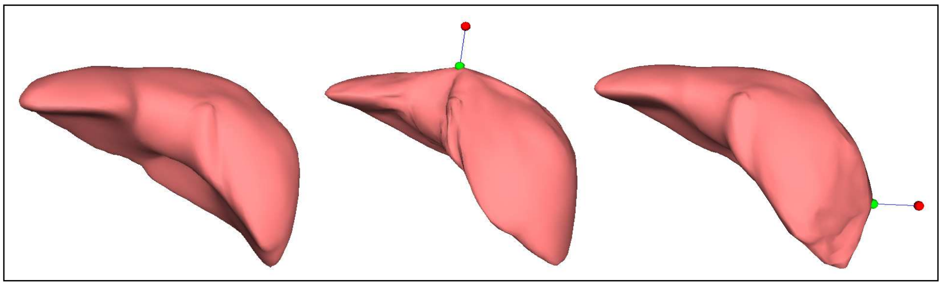

Figure 10 Deforming a virtual liver using IdoLib (mass-spring).

7 Conclusions

This paper reports some recent results obtained in the framework of the center for endoscopic computer-assisted surgery EndoCAS. We have presented some applications of physical simulation to medicine: a method for automatic segmentation and reconstruction of shapes from noisy 3D datasets, a technique to simulate drilling over quasi rigid objects, an approach for the interactive simulation of deformable objects allowing cuts and deformations and a method to simulate surgical threads. The paper also present the library for physical simulation which is being developed and over which the applications described above were built.

References

Agus, M., Giachetti, A., Gobbetti, E., Zanetti, G. and Zorcolo, A. (2002) 'Real-time haptic and visual simulation of bone dissection', IEEE Virtual Reality Conference, pub-IEEE, pp. 209-216.

Agus, M., Gobbetti, E., Pintore, G., Zanetti, G. and Zorcolo, A. (2006) 'Real-time cataract surgery simulation for training', *EuroGraphics Italian Chapter Conference*.

Bielser, D., Maiwald, V. and Gross, M. (1999) 'Interactive cuts through 3-dimensional soft tissue', *Computer Graphics Forum*, *EuroGraphics '99 Proceedings*, 18, 3, C31-C38.

Brown, J., Latombe, J.C. and Montgomery, K. (2004) 'Real-time knot-tying simulation', *The Visual Computer*, vol. 20, num. 2-3, pp. 165-179.

Caselles, V., Kimmel, R. and Sapiro, G. (1995) 'Geodesic active contours', *Fifth International Conference on Computer Vision*.

Cohen, L.D. and Cohen I. (1993) 'Finite element methods for active contour models and balloons for 2d and 3d images', *IEEE Transactions on Pattern Analysis Machine Intelligence*, vol. 15, num. 11, pp. 1131-1147.

Cotin, H.D.S. and Ayache, N. (1999) 'A hybrid elastic model allowing real-time cutting, deformations and force feedback for surgery training and simulation', *CAS99 Proceedings*, pp. 70-81.

Cotin, H.D.S. (1998) 'Efficient Linear Elastic Models of Soft Tissues for real-time surgery simulation', *Technical Report, Istitut National de Recherche en Informatique et en Automatique*.

Costa, I.F. and Balaniuk, R. (2001) 'Static solution for real time deformable objects with fluid inside', *ERCIM News*.

Ganovelli, F., Cignoni, P., Montani, C. and Scopigno, R. (2000) 'Enabling cuts in multiresolution representation', *CGI 2000 Proceedings*.

Ganovelli, F., Cignoni, P., Montani, C. and Scopigno, R. (2000) 'A multiresolution model for soft objects supporting interactive cuts and lacerations', *Computer Graphics Forum*.

Lorensen, W.E. and Cline, H.E. (1987) 'Marching cubes: A high resolution 3D surface construction algorithm', *ACM Computer Graphics*, *SIGGRAPH 87 Proceedings*, vol. 21, p. 163-170.

Malladi, R., Sethian, J. and Vemuri, B. (1995) 'Shape modeling with front propagation: A level set approach', *IEEE Transactions on Pattern Analysis Machine Intelligence*, vol. 17, num. 2, pp. 158-175.

McInerney, T. and Terzopoulos, D. (1996) 'Deformable models, in medical image analysis: A survey', *Medical Image Analysis*, vol. 1, num. 2, pp. 91-108.

McNeely, W.A., Puterbaugh, K.D. and Troy, J.J. (1999) 'Six degree-of-freedom haptic rendering using voxel sampling', *Proceedings of ACM SIGGRAPH*, pp. 401-408.

Morris, D., Sewell, C., Blevins, N., Barbagli, F. and Salisbury, K. (2004) 'A collaborative virtual environment for the simulation of temporal bone surgery', *MICCAI*, *Lecture Notes in Computer Science*, vol. 3217, pp. 319-327.

Müller, M., Heidelberger, B., Teschner, M. and Gross, M. (2005) 'Meshless deformations based on shape matching', *Proceedings of ACM SIGGRAPH*, vol. 24, num. 3, pp. 471-478.

Müller, M., Keiser, R., Nealen, A., Pauly, M., Gross, M. and Alexa, M. (2004) 'Point based animation of elastic, plastic and melting objects', *Proceedings of ACM SIGGRAPH/EUROGRAPHICS Symposium on Computer Animation.*

Müller, M., Heidelberger, B., Hennix, M. and Ratcliff, J. (2007) 'Position based dynamics', *J. Vis. Comun. Image Represent.*, vol. 18, num. 2, pp. 109-118.

Nienhuys, H.W. and van der Stappen, A.F. (2000) 'Combining finite element deformation with cutting for surgery simulations', *EuroGraphics Short Presentations*, pp. 43-52.

O'Brien, J.F. and Hodgins, J.K. (1999) 'Graphical modeling and animation of brittle fracture', *SIGGRAPH*, pp. 137-146.

Pauly, M., Keiser, R., Adams, B., Dutr, P., Gross, M. and Guibas, L.J. (2005) 'Meshless animation of fracturing solids', *ACM Transactions on Graphics*, vol. 24, num. 3, pp. 957-964.

Petersik, A., Pflesser, B., Tiede, U., Höhne, K. and Leuwer, R. (2002) 'Haptic volume interaction with anatomic models at sub-voxel resolution', *Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, pp. 66-72.

Steinemann, D., Otaduy, M.A., Gross, M. (2006) 'Fast arbitrary splitting of deforming objects', *EuroGraphics/SIGGRAPH Symposium on Computer Animation*.

Teschner, M., Heidelberger, B., Müller, M., Pomeranets, D. and Gross, M. (2003) 'Optimized spatial hashing for collision detection of deformable objects', *Proceedings of the Conference on Vision, Modeling and Visualization 2003 (VMV-03)*, pp. 47-54.

Wang, F., Burdet, E., Vuillemin, R. and Bleuler, H. (2006) 'Knot-tying with visual and force feedback for VR laparoscopic training', *IEEE Computer Society*, pp. 5778-5781.