An Investigation of Meshless Deformation for Fast Soft Tissue Simulation in Virtual Surgery Applications

Alex Henriques, Burkhard Wünsche and Stefan Marks

Division for Biomedical Imaging and Visualization, Department of Computer Science, University of Auckland, New Zealand

Abstract. Soft tissue simulation is an important technique in many biomedical modelling applications. While applications for investigating the functioning of organs do require a physically correct mathematical model, other tasks, such as virtual surgery simulations for training purposes, put a bigger emphasis on speed, ease of use, and a large variety of arbitrary interaction. In such applications a plausible, physically not accurate technique is often sufficient. We have investigated the use of meshless deformation based on shape matching for soft tissue simulation and we implemented several improvements which make the simulation more accurate and stable. Novel interaction techniques such as fast, interactive cutting, pushing and picking are introduced. The method is very efficient, easily implemented, and can be integrated into graphics and game engines, but has restrictions in the range of simulated objects.

Keywords: Meshless Deformation, Shape Matching, Soft Tissue Deformation, Virtual Surgery

1. Introduction

Soft tissue simulation is an important technique in many biomedical modelling applications [1]. Existing solutions can be divided into pre-animations, kinematic methods, geometric methods, and physically-based methods.

Pre-animated simulations are achieved by modelling a limited range of interactions using a human animator or more complex physically-based techniques. The simulations are stored in movie or 3D animation formats and are triggered when the user performs certain predefined operations such as cutting in a specified region. This type of simulation does not allow arbitrary interactions, but is fast and can be achieved using game engines, flash animations and other widely available tools. Such simple simulate-ions can be useful in virtual surgery tools teaching process rather than motor skills [2].

Kinematic methods do not represent material properties and forces and include direct mesh manipulation and implicit surfaces. Free-form deformation associates object coordinates with locations in a surrounding mesh of control points [3]. If the control mesh is deformed the object deforms with it. The technique is quite simple, but offers limited forms of manipulations, and makes it difficult to implement cutting operations. Implicit surfaces are defined as isosurface of a 3d density field, which is generated from basic objects by using density functions which are one at the object and zero at a radius of influence away from it. The method results in a computationally easy collision detection and response, but is expensive to render, difficult to integrate into graphics engines and it is hard to control shapes and simulate cutting operations.

We use the term geometric methods for techniques which use physical properties to kinematically deform regions of an object's geometry. Delp et al. [5] represent different types of tissue using polygonal surface meshes. When interacting with the tissue, the nearest contact point to the surgical instrument is calculated. Affected vertices in a predefined area around this contact point are deformed using a polynomial interpolation.

Physically-based methods include mass-spring systems and finite element methods. Mass-spring systems represent tissue as a set of points where neighbouring points are connected by springs which simulate the elasticity of the tissue. The method is easy to implement and cutting can be modeled by removing springs [6]. Finite element simulations model the volumetric nature of soft tissue and describe its deformation behaviour using a set of differential equations incorporating material parameters [7]. The resulting simulations are physically realistic but are computationally expensive. Point-Associated Finite Field (PAFF) uses a multi-resolution representation where non-linear responses are only considered in the immediate vicinity of the applied force in order to obtain real-time non-linear deformations [8].

All of the above methods are either computationally expensive or only offer a limited number of interactions and are difficult to implement and integrate into commonly available graphics engines.

2. Improved meshless deformation

Meshless Deformation Based on Shape Matching, or meshless deformation for short, was developed by Müller et al. as a technique for dynamically simulating deformable objects [9]. The approach is geometrically motivated, requires no mesh connectivity and pre-processing, and is fast, simple to compute and unconditionally stable.

Meshless deformation matches a set of points defining an object to a goal configuration. To preserve volume, meshless deformation scales the deformation matrix defining the transformation. However, the image sequence at the top of figure 1 demonstrates that as a result an object squashed along one dimension is scaled up drastically in the other two dimensions in response.

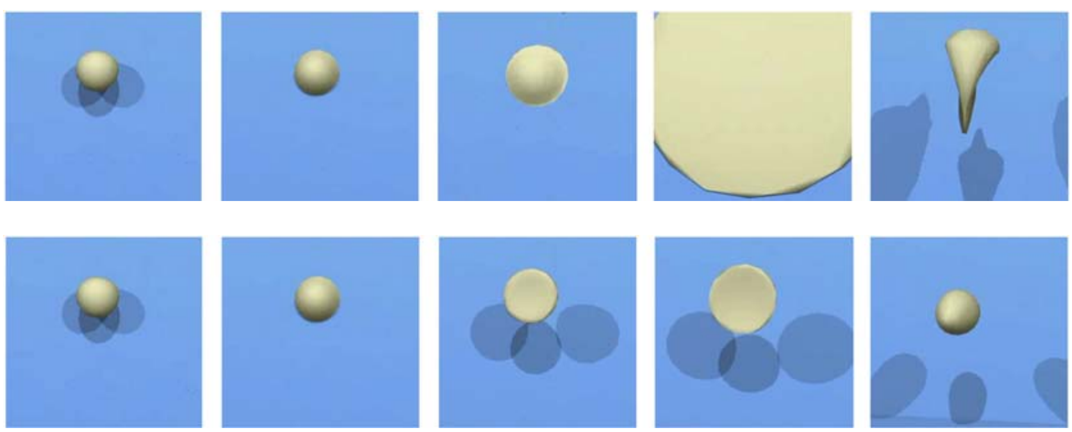

Fig. 1. A ball smashed with high force against a wall. The original meshless deformation algorithm (top) is unconditionally stable but leads to unrealistic intermediate configurations. Our modification (bottom) limits the scale and shear factors and results in more realistic looking results.

We develop a solution to this problem based on the observation that the deformation of a fluid filled object is limited because its surface offers resistance towards the deforming forces. In previous work we investigated several solution methods [10] and we found that the most promising approach is to limit the deformation by increasing the rigidity of the object, which is achieved by putting a cap on the Frobenius norm of the matrix containing the non-rigid component of the object's transformation. In order to prevent discontinuities in the motion we use a monotonically increasing weighting function which uses the original deformation matrix for small deformations and increases the weighting of the rigid component to one when the Frobenius norm of the non-rigid deformation approaches the predefined limit. The results are illustrated in the image sequence at the bottom of figure 1. The effect of our modification is similar to the non-linear stress-strain relationship in biological tissues such as skin where deformations are linear for small forces, but do not exceed a certain limit until the tissue breaks.

3. Interaction techniques

An important component of virtual surgery simulations is the interaction with tissue. We have implemented picking, constraining, pushing and cutting of objects. The picking mode allows the user to grab an object and to move it around. We use the closest object vertex as pick point and connect it via a spring to the user's cursor. The user's mouse movements are interpreted as 2D motion in the plane parallel to the view plane and the mouse wheel is used to move the cursor orthogonal to the view plane. More direct interaction with an object can be achieved by increasing the spring force. Springs can be locked into place allowing the user to fix tissue into certain positions.
The pushing mode allows the user to deform or push objects aside. Collision response forces are applied to all objects near the cursor.
The cutting tool is represented by two "blades" defining a cutting plane which the user can orientate arbitrarily. Since meshless deformation does not require mesh connectivity cutting is implemented efficiently using a simple clipping operation. Surface meshes for rendering are obtained using a Delaunay triangulation along the cutting surface [10].

4. Results

We have developed a framework for testing interactive simulation environments and implemented within it meshless deformation based on shape matching together with our improvements. We found that simple objects with limited modes of deformation are simulated best, while objects composed of simple subcomponents are simulated well with clusters. Good results were obtained when applying large deformations to blobby objects such as kidney shaped models and convoluted tube like structures. In contrast facial animation failed because human are intimately familiar with facial emotions and any deviations from physical accuracy are clearly noticeable.

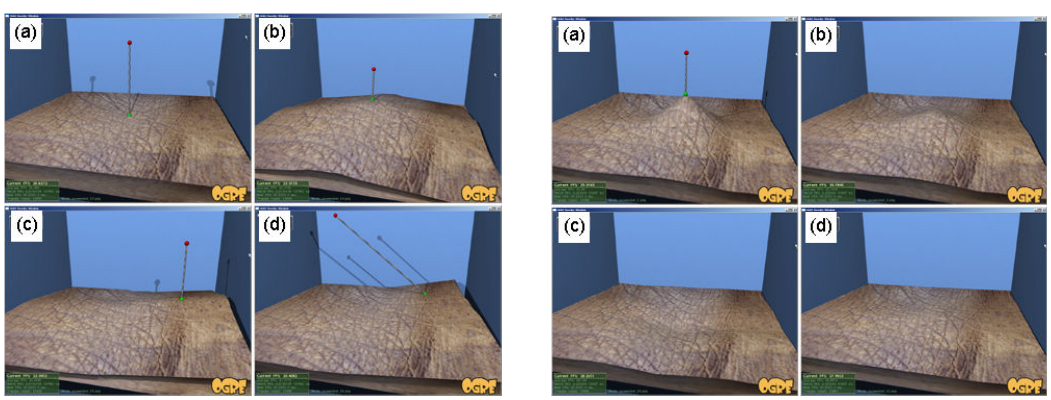

Fig. 2. Simulation of skin. Left: using a single cluster (a) and 2x2 clusters (b-c). Right: using 5x5 clusters.

Advantages of our implementation are the high speed which enables us to deform dozens of soft objects simultaneously, the ease of use, the automatic volume preservation, the immersive feel especially when interacting with and cutting objects, and that the representation can be easily integrated into game engines. No special data formats are required and standard surface meshes can be used. The "gooeyness" and stiffness of objects can be controlled with two parameters defining the deformation [10]. Disadvantages are the lack of physical realism and the lack of local deformation. For example, when simulating a skin patch as shown in figure 2 we need 5x5 patches to get realistic results. The cutting operation is currently restricted to planar cuts and more complex cuts or partial incisions are not yet possible.

5. Conclusion

Meshless deformation based on shape matching is a promising technique for use in interactive and collaborative virtual surgery environments. It is easy to implement and requires no special data formats. As a result it can be easily integrated into game engines in order to develop collaborative virtual surgery applications. This makes it possible to use the full power of such an engine, e.g. network play, superior graphics and a large user base testing and extending it. Furthermore, since no special hardware and software is required any application using this technique can be widely distributed and could over time develop a large user community contributing to it. Preliminary user testing indicates that our interaction tools are easy to use and increase the immersive feel. Meshless deformation is extremely fast and stable and works even for complex environments with dozens of objects. Current disadvantages include the lack of visual realism, the limited number of deformation modes and the need to use small clusters for simulating local deformations. In future work we want to investigate hybrid systems and multi-resolution representations in order to alleviate the current disadvantages of this technique.

References

[1] Liu, A., Tendick, F., Cleary, K. and Kaufmann, C. (2003), A survey of surgical simulation: applications, technology, and education, Presence: Teleoperators and Virtual Environments 12(6), pp. 599–614.

[2] Dunkin, B., Adrales, G. L., Apelgren, K., Mellinger, J. D. (2007), Surgical Simulation: A Current Review, Surgical Endoscopy 21 (3), pp. 357-366.

[3] Sederberg, T. W., Parry, S. R. (1986), Free-form deformation of solid geometric models, ACM SIGGRAPH Computer Graphics 20 (4), pp. 151–160.

[4] Cani-Gascuel, M., Desbrun, M. (1997), Animation of deformable models using implicit surfaces, IEEE Transactions on Visualization and Computer Graphics 3(1), pp. 39–50.

[5] Delp, S., Loan, J., Basdogan, C., Rosen, J. (1997), Surgical simulation: an emerging technology for emergency medical training, Presence: Teleoperators and Virtual Environments 6, pp. 147–159, URL: citeseer.ist.psu.edu/delp97surgical.html.

[6] Kühnapfel, U., Cakmak, H. K. and Maaß, H. (2000), Endoscopic surgery training using virtual reality and deformable tissue simulation, Computers & Graphics 24, pp. 671–682.

[7] Bro-Nielsen, M. (1998), Finite element modeling in surgery simulation, Proceedings of the IEEE 86 (3), 490–503.

[8] De, S., Lim, Y.-J., Manivannan, M. and Srinivasan, M. A. (2006), Physically realistic virtual surgery using the point-associated finite field (PAFF) approach, Presence: Teleoperators and Virtual Environments 15(3), pp. 294–308.

[9] Müller, M., Heidelberger, B., Teschner, M., and Gross, M. (2005). Meshless deformations based on shape matching, ACM Transactions on Graphics 24 (3), pp. 471–478.

[10] Henriques, A., Wünsche, B. (2007). Improved Meshless Deformation Techniques for Real-Time Interactive Collaborative Environments, Proceedings of GRAPP '07, Barcelona, Spain, pp. 154-161.