Research Journal of Applied Sciences, Engineering and Technology 5(17): 4240-4245, 2013
DOI:10.19026/rjaset.5.4411
ISSN: 2040-7459; e-ISSN: 2040-7467

Submitted: June 28, 2012 Accepted: August 08, 2012 Published: May 01, 2013

Research Article

Optimized Real Time Vertex Based Deformation Using Octree and Two Neighborhood Method

Irshad Ahmad, Suziah Bt. Sulaiman and Dayang Rohaya Bt Awang Rambli
Department of Computer and Information Sciences, Universiti Teknologi PETRONAS, Bandar Seri Iskandar, 31750 Tronoh, Perak, Malaysia

Abstract: Soft tissue simulation is very important in medical simulation and learning procedures. But such simulations require intensive computation. With the force feedback devices, the computation required should be much faster as touch sensation is approximately 20 times faster than that of visual. Efficient collision detection techniques are required to quickly locate the touched node of the model and a few triangles of the model need to be rendered in the haptic loop to get further optimization while achieving the same haptic sensation. In this study an octree space partitioning method is used for collision detection to find the touched node quickly and two circular rings of neighbors of the touched node are rendered in the haptic loop for further optimization. This technique is implemented in our previously developed real time vertex based deformation. The results are compared with the previous method which shows better performance.

Keywords: Deformation, interactive, modified slope intercept form, Octree Space Partitioning

INTRODUCTION

Real time interaction with deformable objects is one of the interesting and challenging fields in Computer Graphics. In recent years, with the augment of force feedback devices the interaction with deformable objects became more sensitive and meaningful but at the cost of extensive computation as its refresh rate is much higher than visual. For real time interactive simulation, the algorithm needs to meet the requirement of both visual and haptic interaction in terms of efficiency.

In deformation modelling of soft objects, two methods are used. Following sections discusses these types with main focus on interaction with these deformable models using force feedback device.

Geometry-based: In this method, the geometry of the model is manipulated directly upon interaction. The model represented using this method have no solution for representing the internal physical properties of the model. The advantage of these methods is that the computation is less and the algorithms are easy to be implemented. Two approaches in geometry based are vertex based and Spline based used for deformation (Basdogan, 1999).

- **Vertex based:** In this method, while deforming an object visually, the vertices of the body are manipulated directly. All the points in the region of interest are deformed in such a way that shows visually appealing deformation. This is the most basic and efficient method of visual deformation.
  In Basdogan *et al*. (1998), the author used vertex based deformation of the second order polynomial. In order to deform the soft tissue locally, they translated all vertices within a certain range, called the radius of influence, of the collision point along the surgical instrument. The magnitude of translation is controlled by the second order polynomial and the shape of the deformation is controlled by the degree and the coefficients of the polynomial. This method is only applied for visual deformation.
- **Spline based:** In this method, the object to be deformed is embedded in a linear cube of the grid. This grid acts as a handle for the deformation of an object. By changing the handle position, deforms the object embedded in it.

The pioneering method in this field is proposed in (Sederberg and Parry, 1986). Different variations of this method are proposed in Davis and Burton (1991), Lamousin and Waggenspack (1994), Griessmair and Purgathofer (1989), Hsu *et al*. (1992) and Song and Yang (2005). The author in Hui *et al*. (2006) combined this method with a mass spring model to incorporate physical behaviour of the system.

Physics-based: These models simulate the physical behaviour of objects and consider the internal and

Corresponding Author: Irshad Ahmad, Department of Computer and Information Sciences, Universiti Teknologi PETRONAS, Bandar Seri Iskandar, 31750 Tronoh, Perak, Malaysia

external forces. These methods are more appropriate in soft tissue modelling and therefore have been extensively used in the medical application such as training. In physics based, there are three popular methods for simulating deformable models in real time simulation.

- **Finite element method:** Basdogan (1999), FEM is considered one of the accurate methods for simulation of soft tissue deformation, but is computationally very slow and not suitable for real time interaction including force feedback devices with soft tissue in its pure form. To overcome the computational complexity and make it suitable for real time simulation, many optimization techniques have been proposed (Bro-Nielsen and Cotin, 1996; Vigneron *et al*., 2004; Hauser *et al*., 2003; Yan *et al*., 2007; Picinbono *et al*., 2000; Hadrien *et al*., 2010).
  FEM is used both for linear and non-linear elastic models. Linear FEM is used in Frank *et al*. (2001), Mor and KanAde (2001), Nienhuys and Van Der Stappen (2000), Lindblad and Turkiyyah (2007), Sela *et al*. (2007) and Wu and Heng (2005); while in Picinbono *et al*. (2003) non-linearity is implemented in the contact region of soft tissue.
- **Mass spring systems:** A Mass Spring System is a popular approach in Physics Based deformation. The reason for popularity is its efficiency and ease of implementation. A Mass Spring model considers continuous object as a finite set of discrete mass points, also known as nodes. These nodes are then connected with each other through massless springs forming a lattice. The deformation occurs when the nodes are displaced by some external forces such gravity or user applied forces and internal forces in the form of spring force. Stronger the spring force, the stiffer is the object it represents. Various numerical methods are used for calculating the new position of nodes such as Euler, Backward difference method, *Runge kutta* order 4.

After the pioneering work done by Terzopoulos *et al*. (1987) and Terzopoulos and Fleischer (1988), this approach has been used for cloth simulation (Baraff and Witkin, 1998; Provot, 1995), face animation (Kahler *et al*., 2001) and importantly for soft tissue behaviour modelling in surgery training simulator (Mollemans *et al*., 2004; Zhang *et al*., 2005; Brown *et al*., 2002; Choi *et al*., 2003).

A big challenge in MSS is setting stiffness parameters. Recent research is going on addressing the parameter finding for the mass spring model. Various techniques have been proposed in the literature for parameter finding as in Etzmuss *et al*. (2003), Bianchi *et al*. (2003), D'Aulignac *et al*. (1999), Baran and Basdogan (2010) and Natsupakpong and Çavusoglu (2010).

Although many techniques for deformable modelling exist, there is no one technique which has all of the characteristics i.e. Speed, robustness, physiological realism and topological flexibility which

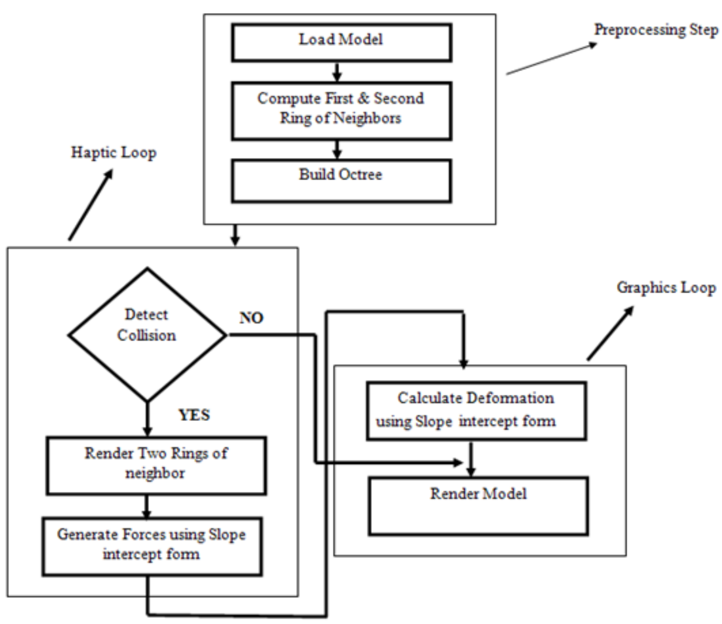

Fig. 1: Haptic and graphics loop synchronization

needs of Virtual reality applications such as surgical simulation (Meter *et al*., 2005).

A hybrid method is presented in Ahmad and Sulaiman (2011) which combines vertex based deformation with the Mass spring systems. In this method the nodes of the model are directly deformed using slope intercept form of a line equation for the touched and its two rings of neighbours. The touched node and its two circular rings are deformed using different slope values. These deformed nodes are attached to the fixed nodes using a spring. Once the force is relaxed the deformed nodes are attracted towards fixed nodes using these springs. To feel the object softness the same spring reaction force is rendered to the user using a force feedback device. The flow of the simulation is given in Fig. 1.

In this study we optimized our previously developed algorithm for Visio haptic deformation (Ahmad and Sulaiman, 2011). This optimization is twofold, one using octree space partitioning method to partition the model space and use this for fast accessing to the touched node on the model. Second once found the touched node in the model, render its two rings of neighbours in the haptic loop to feel the object. This greatly improves the haptic cycle efficiency.

METHODOLOGY

Visio - Haptic Deformation: In real time vertex based deformation (Ahmad and Sulaiman, 2011), the vertex is deformed based upon the slope intercept form of a line equation. The user touches the model with the Phantomforce feedback device (SensAble Technologies, http:// www.sensable.com)).

Deformation is achieved by directly deforming the nodes of the model. Once touched with the force feedback device, this touched node and its two rings of neighbours needs to be deformed in order to get realistic deformation with different slopes.

Touched node is deformed using slope intercept form using Eq. (1):

$$p_{i+1}(x,y,z) = p_i(x,y,z) + m_1 F(x,y,z) + b_1 \qquad (1)$$

Here b_1 is set to zero and m_1 must be less than zero. p_{i+1} are the new position of the node p and p_i is the old position. F is the force applied on the node p_i through a haptic force feedback device.

Now the two circular rings of this touched node are deformed using the same slope intercept form of a line equation but with different slopes.

The first ring of neighbours is deformed using Eq. (2) and second ring of neighbour is deformed using Eq. (3). The deformation profile is shown in Fig. 2:

$$pN_{i+1}(x,y,z) = pN_i(x,y,z) + m_2 F(x,y,z) + b_2 \qquad (2)$$

$$pNN_{i+1}(x,y,z) = pNN_i(x,y,z) + m_3 F(x,y,z) + b_3 \qquad (3)$$

In (1), (2) and (3), $b_1 = b_2 = b_3 = 0$ and m_1, m_2 and m_3 are negative and in the range of [0,-1], exclusive of 0 and -1 and must satisfy the condition: $m_1 > m_2 > m_3$.

Searching: Octree Space Partitioning (OSP) method is used to partition the space occupied by the model. A model is made up of nodes, so the nodes are partitioned and stored in the leaves of the octree. This is shown in Fig. 3.

When the user touches the model using force feedback device then searching is much faster as compared to the linear search using the nearest neighbour method available in the OpenHaptics APIs. This method is used to locate the touched node quickly. Algorithm for searching node is:

searchNode (OSP octreeNode, HCP Point, int* touchedNode)

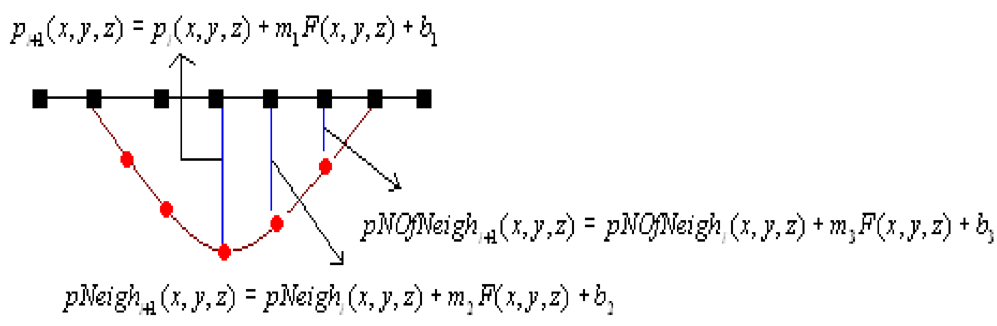

Fig. 2: Deformation profile using a modified slope intercept form

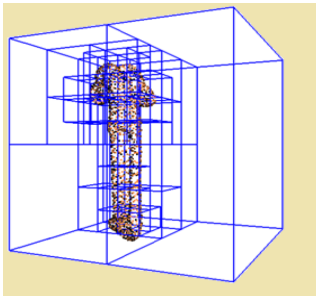

Fig. 3: Model nodes are partitioned using OSP

```
{
if (octreeNode =NULL)
    return;
if (Point outside the boundaries of octreeNode)
return;
if (octreeNode is Subdivided)
for ( i = 0; i<8; i++)
{
call searchNode( octreeNode->child[i], Point, & touchedNode)
}
else
{

Loop through nodes in octreeNode->child[touched]
Find closest node to the point
touchedNode = touchedNodeIndex
}
}
```

Rendering: Once found the touched node of the model, the next step is to render its two circular neighbors in the haptic rendering loop instead of rendering the whole model. This rendering of two rings of neighbours is visualized during simulation for touching various nodes of the same model in the Fig. 4

An algorithm for rendering two rings of neighbours:

```
void drawinHapticLoop()
{
for (int i = 0; i<nodes[touchedNode]. numfirstRing; i++)
{
Render touchedNode first and second ring of neighbors
}
```

This function executes in the haptic loop and render only two circular rings of the touched node.

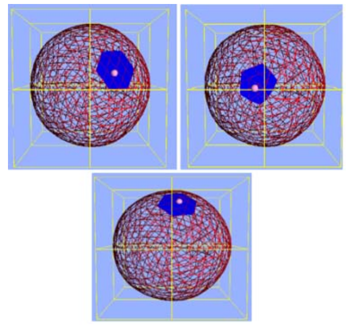

Fig. 4: Rendering two rings of neighbours in the haptic loop for various touched nodes

Table 1: Rendering time for the previous and proposed method for different model resolution

Model resolution		Time taken in microseconds (Haptic rendering loop) Previous method	Hybrid method	Proposed method
Vertices:	1442	4.8	3.8	0.90
Triangles:	2880			
Vertices:	3970	13.50	10.10	1.40
Triangles:	7937			
Vertices:	5762	19.50	12.50	1.85
Triangles:	11521			
Vertices:	8664	30.10	21.50	2.70
Triangles:	17325			

EXPERIMENTAL RESULTS

We compare our new optimized real time vertex based deformation with the previously developed algorithm. The time required for rendering in the haptic loop for the previous and proposed method is given in Table 1, for different resolutions of the same model. Table 1, shows the time required for different resolution models using the previous method, hybrid method and the proposed method. Previous method has linear searching using nearest neighbour method available in the OpenHaptics API and the whole model is rendered in the haptic loop. Hybrid method has octree for searching the touched node and the whole model is rendered in the haptic loop. Proposed method uses octree for touching node searching and rendering two rings of neighbours in the haptic loop. All these mentioned methods use vertex based deformation using slope intercept of the line equation. Deformation using the proposed method is shown in Fig. 5 for stomach model and in Fig. 6, for simple sphere.

Table 1, shows the proposed method has higher performance as compared to our previous method as well as with the hybrid method containing an octree method for searching.

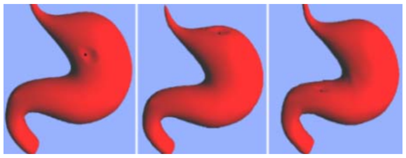

Fig. 5: Interactive deformation of stomach model

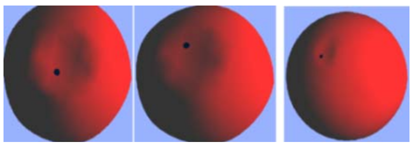

Fig. 6: Interactive deformation of sphere model

CONCLUSION

In this study, an improved algorithm for visual deformation based on a modified slope intercept form of a line equation is presented. The algorithm is fast in visual deformation without needing efficient data structures for mesh. This algorithm works for any number of nodes in the mesh. Our algorithm is best for linear, isotropic and homogenous soft objects. To get realistic deformation of the complex mesh, rings of neighbours should need to be increased. The algorithm shows much improvement in terms of speed for increasing model resolution and more suitable for real time interactive simulators involving touch sensation. We applied our algorithm to surface based meshes. In further study we will apply our algorithm to volumetric meshes as well.

REFERENCES

Ahmad, I. and S.B. Sulaiman, 2011. Real time vertex based deformation in training simulator. Proceeding of International Symposium on Ubiquitous Virtual Reality (ISUVR), pp: 21-24.

Baraff, D. and A. Witkin, 1998. Large steps in cloth simulation. Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '98). ACM New York, USA, pp: 43-54.

Baran, B. and C. Basdogan, 2010. Force-based calibration of a particle system for realistic simulation of nonlinear and viscoelastic soft tissue behavior. EuroHaptics, LNCS 6191, Part 1, pp: 23-28.

Basdogan, C., 1999. Force reflecting deformable objects for virtual environments in haptics: From basic principles to advanced applications. Proceeding of 26th International Conference on Computer Graphics and Interactive Techniques (SIGGRAPH' 99). Course No: 38, August 8-13, Los Angeles.

Basdogan, C., C. Ho and M.A. Srinivasan, 1998. Force interaction in laparoscopic simulation: Spline based haptic rendering soft tissues. Medicine Meets Virtual Reality, VI, San Diego, CA, January 19-22, pp: 385-391.

Bianchi, G., M. Harders and G. Székely, 2003. Mesh Topology Identification for Mass-spring Models. In: Ellis, R.E. and T.M. Peters (Eds.), MICCAI, LNCS, Springer, Heidelberg, 2878: 50-58.

Bro-Nielsen, M. and S. Cotin, 1996. Real-time volumetric deformable models for surgery simulation using finite elements and condensation. Comput. Graph. Forum, 15(3): 57-66.

Brown, J., S. Sorkin, J.C. Latombe, K. Montgomery and M. Stephanides, 2002. Algorithmic tools for real-time microsurgery simulation. Med. Image Anal., 6(3): 289-300.

Choi, K.S., H. Sun and P.A. Heng, 2003. Interactive deformation of soft tissues with haptic feedback for medical learning. IEEE T. Inf. Technol. B., 7(4).

Davis, O.R. and R.P. Burton, 1991. Free-form deformation as an interactive modeling tool. J. Imag. Technol., 17(4): 181-187.

D'aulignac, D., M.C. Cavusoglu and C. Laugier, 1999. Modeling the Dynamics of the Human Thigh for a Realistic Echographic Simulator with Force Feed-back. In: Taylor, C. and A. Colchester (Eds.), MICCAI, Springer, Heidelberg, 1679: 1191-1198.

Etzmuss, O., J. Gross and W. Strasser, 2003. Deriving a particle system from continuum mechanics for the animation of deformable objects. IEEE T. Vis. Comput. Gr., 9(4): 538-550.

Frank, A.O., I.A. Twombly, T.J. Barth and J.D. Smith, 2001. Finite element methods for real-time haptic feedback of soft-tissue models in virtual reality simulators. Proceedings of the Virtual Reality Conference (VR'01), pp: 257-263.

Griessmair, J. and W. Purgathofer, 1989. Deformation of solids with trivariate B-splines. Proceedings of Eurographics, Elsevier, North Holland, pp: 137-148.

Hadrien, C., J. Hoeryong, A. Jérémie, D. Christian, D.Y. Lee and C. Stéphane, 2010. GPU-based real-time soft tissue deformation with cutting and haptic feedback. Prog. Biophys. Molecul. Biol., 103(2-3): 159-168.

Hauser, K.K., C. Shen and J.F. O'Brien, 2003. Interactive deformation using modal analysis with constraints. Proceedingsof Graphics Interface, pp: 247-256.

Hsu, W.M., J.F. Hughes and H. Kaufman, 1992. Direct manipulation of free-form deformations. SIGGRAPH '92: Proceedings of the 19th Annual Conference on Computer Graphics and Interactive Techniques. ACM Press, pp: 177-184.

Hui, C., S. Hanqiu and J. Xiaogang, 2006. Interactive Haptic Deformation of Dynamic Soft Objects. VRCIA, Hong Kong, pp: 14-17.

Kahler, K., J. Haber and H.P. Seidel, 2001. Geometry-based muscle modeling for facial animation. Proceeding of Graphics Interface Conference, pp: 37-46.

Lamousin, H. and W. Waggenspack, 1994. NURBS-based free-form deformation. IEEE Comput. Graph., 14(9): 59-65.

Lindblad, A. and G. Turkiyyah, 2007. A physically-based framework for real-time haptic cutting and interaction with 3d continuum models. Proceedings of the ACM Symposium on Solid and Physical Modeling (SPM ’07), pp: 421-429.

Meter, U., O. Lopez, C. Monserrat, M.C. Juan and M. Alcantz, 2005. Real-time deformable models for surgery simulation : A survey. Comp. Meth. Prog. Biomed., 77: 183-197.

Mollemans, W., F. Schutyser, J.V. Cleynenbreugel and P. Suetens, 2004. Fast soft tissue deformation with tetrahedral mass spring model for maxillofacial surgery planning systems. Proceeding of Medical Image Computing and Computer-Assisted Intervention (MICCAI ’04), pp: 371-379.

Mor, A.B. and T. KanAde, 2000. Modifying soft tissue models: Progressive cutting with minimal new element creation. Proceeding of Medical Image Computing and Computer-Assisted Intervention (MICCAI), pp: 598-608.

Natsupakpong, S. and M.C. Çavusoglu, 2010. Determination of elasticity parameters in lumped element (mass-spring) models of deformable objects. Graph. Mod., 72: 61-73.

Nienhuys, H.W. and A.F. Van Der Stappen, 2000. A Surgery Simulation Supporting Cuts and Finite Element Deformation. Springer Berlin, Heidelberg, pp: 145-152.

Picinbono, G., H. Delingette and N. Ayache, 2003. Non-linear anisotropic elasticity for real-time surgery simulation. Graph. Mod., 65(5): 305-321.

Picinbono, G., J.C. Lombardo, H. Delingette and N. Ayache, 2000. Anisotropic elasticity and force extrapolation to improve realism of surgery simulation. Proceeding of IEEE International Conference on Robotics and Automotion (ICRA), pp: 596-602.

Provot, X., 1995. Deformation constraints in a spring-mass model to describe rigid cloth behavior. Proceeding of Graphics Interface Conference, pp: 147-154.

Sederberg, T. and S. Parry, 1986. Free-from deformation of solid geometric models. Comput. Graph., 20(4): 151-160.

Sela, A.G., J. Subag, A. Lindblad, D. Albocher, S. Schein and G. Elber, 2007. Real-time haptic incision simulation using fem-based discontinuous free-form deformation. Comput. Aided Des., 39(8): 685-693.

Song, W. and X. Yang, 2005. Free-form deformation with weighted T-spline. Visual Comp., 21: 139-151.

Terzopoulos, D. and K. Fleischer, 1988. Modeling inelastic deformation: Viscolelasticity, plasticity, fracture. Proceedings of the 15th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '88). ACM, New York, USA, pp: 269-278.

Terzopoulos, D., J. Platt, A. Barr and K. Fleischer, 1987. Elastically deformable models. SIGGRAPH Comp. Graph., 21(4): 205-214.

Vigneron, L.M., J.G. Verly and S.K. Warfield, 2004. Modelling Surgical Cuts. Retractions and Resections Via Extended Finite Element Method. Springer Berlin, Heidelberg, pp: 311-318.

Wu, W. and P.A. Heng, 2005. An improved scheme of interactive finite element model for 3d soft tissue cutting and deformation. Visual Comp., 21(3): 707-716.

Yan, Z., L. Gu, P. Huang, S. Lv, X. Yu and X. Kong, 2007. Soft tissue deformation simulation in virtual surgery using nonlinear finite element method. Conf. Proc. IEEE Eng. Med. Biol. Soc., 2007: 3642-3645.

Zhang, S., L. Gu1, P. Huang and J. Xu, 2005. Real-time simulation of deformable soft tissue based on mass-spring and medial representation. Proceeding of 1st Int’l Workshop Computer Vision for Biomedical Image Applications (CVBIA ’05), pp: 419-426.