

Master Project - Real Time Rendering of skeletal structures - Motivations Goals and Plan

OLIVIER ROUILLER - S090842
Department of Informatics and Mathematical Modelling
Technical University of Denmark

February 28, 2011

1 Related Work

1.1 Skeletal modeling

Surfaces defined by skeletal structures have been popular in Computer Graphics, applications in modelling are possible, mesh generation of organic structures such as trees, blood vessels have been done using skeletal structures. A skeletal structure can easily be generated with L-Systems.

In addition, modelling at a polygon level requires expert skills and is time consuming and a more compact representation of the models is useful to make the modelling easier and is interesting for procedural generation of content.

Bloomenthal [Blo95] used the idea that shapes are defined by their skeleton to model organic objects such as hands, leafs and trees. Implicit surfaces are suitable to define the geometry from a skeleton since their topology is robust. The surface is represented as a convolution surface. Convolution surfaces offers large possibilities to create surfaces from discrete primitives. They have been studied thoroughly in Andrei Sherstyuk's thesis in [She99]. This representation allows to define a smooth and topologically robust surface to primitive such as points, segments, curves, polygons or even meshes. These primitive are convoluted with distance functions such as Gaussians or other metaballs function and these fields are added or combined in different ways for blending.

Wyvil et al. designed a data structure, the BlobTree to store these primitives and operations. [WGG98].

This approach have been studied for modelling of organic structures or characters.

1.2 Skeletal meshing

However generating meshes from implicit surfaces is hard. The most common algorithm is marching cubes [LC87] but it can fail to reconstruct small features if the resolution of the grid is too low and triangles are not aligned to the shapes's features. In addition, it generates a large number of polygons. Some works have been done to use curvature information to reconstruct an implicit surface [AJ04] but the results are not good enough to be a viable alternative to marching cubes (more?).

Other approaches have been studied to generate meshes from skeletal structure, a class of algorithm related to tube surfaces ([FWB04] and [SE05]). In this approach, a tube mesh is grown recursively along the skeleton, particular care is given at branchings.

More recently, algorithms have been developed to generate low detail, feature aligned quad mesh from a skeleton and a set of control points with radius ([JLW10] and Andreas). The algorithm presented in [JLW10] first meshes the skeleton in a way similar that the one described in [SE05] and then the mesh is improved using the implicit field defined by the spheres seen as metaballs, curvature information and optimization is used to align the edges to the shape features.

Curvature lines are indeed attractive when dealing with the placement of vertices and edges of a mesh and work have been done in remeshing using these features [ASD⁺03].

Convolution and implicit surfaces appear to be an appealing representation of a skeletal structure in the modelling perspective but meshing them in a nice way is still to be done.

1.3 Skeletal ray tracing

However, the implicit nature of this surfaces allows us to raytrace them. This technique was only suitable for offline rendering but advances in graphic hardware allowed to perform ray tracing in real time on the GPU. The main applications have been for fluid rendering and simulation, but it is very likely that GPU ray tracing becomes part of the graphics pipeline for general rendering in the game industry and in other fields of visualization.

Recent works presented methods to ray trace efficiently implicit surfaces on the GPU. Loop renders implicit surface by using Bezier Clipping [LB06], encoding the implicit function in a Bezier curve defined at the vertices of tetrahedra. This allows fast and exact root finding for iso surface extraction. [FNN94] reuses Bezier clipping in addition with depth peeling to render a very large numbers of metaballs in real time. Finally, [GPP⁺10] uses a bounding volume hierarchy on the GPU.

2 Motivations

It is likely that the geometric information of a skeleton enables optimization to ray trace skeletal based metaballs as well as more general convolution surfaces. Our work would be to investigate efficient ways to store and render skeletal defined convolution surfaces. To prove the method, we would develop a skeleton structure to which we can attach different convolution surfaces, enough to be able to model appealing models and develop an optimized algorithm to ray trace it on the GPU.

Then to prove the potentiality of our method to be an alternative to polygon rendering, we would need to demonstrate some classic effects (shadowing, ambient occlusion, NPR, ???). Also with a mesh representation, an artist can add details by sculpting, texturing, displacement mapping, normal mapping,... Sculpting can be done by editing the implicit function but how to save this editing? Texturing, displacement mapping and normal mapping basically reduces to texturing.

3 Goals

- Design and implement a data structure to store the convolution surface (need to choose the kind of primitives and convolution kernels).
- Use this representation in a new algorithm to retrace it on the GPU. Must not be a lot worse than polygon rendering.
- Think about ways to enable advanced texturing.
- Prove that it is a viable way of rendering by showing basic animation and some effects.

4 Risks

- Algorithm not fast enough.
- Difficulty to get good model.
- Animation might not be nice with surface blending.
- Advanced texturing may just be impossible.
- Algorithm may not be compatible with some effects.

5 Work done so far

- Simple implicit-skeletal structure I implemented a simple tree structure with position and radius informations, the tree is read and converted to an implicit function with metaballs and metatubes, i.e. convolution surfaces defined by points and segments. Convolution kernel quartic polynomial $h(r) = (R - r^2)^2, R < 1, 0, r > R$, r being the distance to the discrete primitive (point or segment). Implicit surface polygonized with marching cubes.

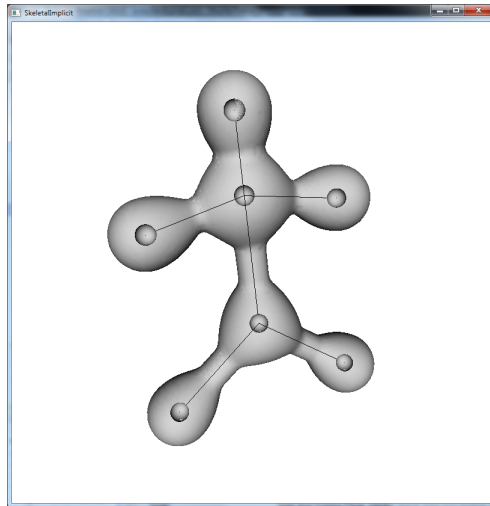


Figure 5.1: Implicit Skeleton polygonized

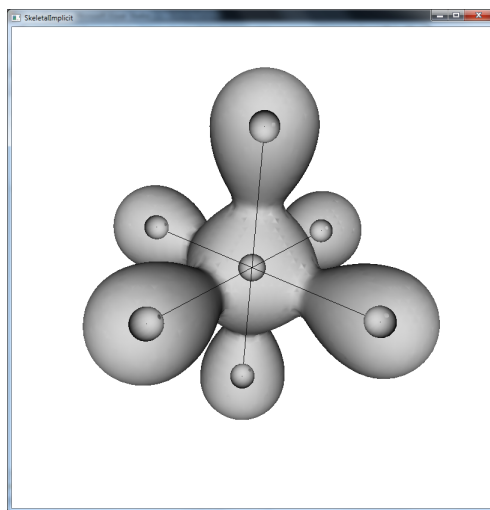


Figure 5.2: Implicit Skeleton polygonized

- Brute force ray tracing I rendered a cube and ray-traced on the fragment shader 2 metaballs and a metatube, naive raymarching and root finding.

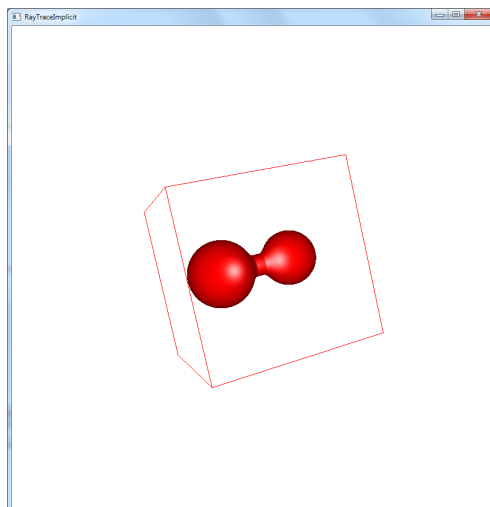


Figure 5.3: 2 metaballs and a metatube raytraced on the GPU

February				March			
W5	W6	W7	W8	W9	W10	W11	W12
Skeletal structure Surface representation Visualisation by tesselation Draft of skeletal data structure Raytracing of skeletal structure with balls and tubes Raytracing of a single metaball Raytracing of a single metatube	Redisgn of the skeletal structure and editor More primitives and convolution	Continue on the structure	Design of Ray tracing algo	DADIU PRODUCTION!			
				RayTracing optimization	RayTracing optimization	RayTracing optimization	
April				May			
W13	W14	W15	W16	W17	W18	W19	W20
RayTracing optimization	RayTracing optimization	Improve shader	Design texturing method	Texturing	Texturing	Animation of the tree structure Might broke raytracing and texturing	Fix
		Texturing					
		Design texturing method					
June				July			
W21	W22	W23	W24	W25	W26	W27	W28
Fix	Integrate in classical pipeline	Shader effects	Shader effects	Shader effects	Shader effects	Writting Polishing	Writting Polishing
August							
W29	W30						
Writting Polishing	Writting Polishing						

References

- [AJ04] Bruno Rodrigues de Araujo and Joaquim Armando Pires Jorge. Curvature dependent polygonization of implicit surfaces. In *Proceedings of the Computer Graphics and Image Processing, XVII Brazilian Symposium*, pages 266–273, Washington, DC, USA, 2004. IEEE Computer Society.
- [ASD⁺03] Pierre Alliez, David C. Steiner, Devillers Devillers, Bruno Lévy, and Mathieu Desbrun. Anisotropic polygonal remeshing. In *Proceedings of ACM SIGGRAPH 2003*, volume 22(3) of *ACM Transactions on Graphics*, pages 485–493. ACM Press, 2003.
- [Blo95] Jules Bloomenthal. *Skeletal design of natural forms*. PhD thesis, Calgary, Alta., Canada, Canada, 1995. UMI Order No. GAXNN-03088.
- [FNN94] Tomoyuki Nishita Fukuyama, Tomoyuki Nishita, and Eihachiro Nakamae. A method for displaying meta-balls by using bezier clipping. *Computer Graphics Forum*, 13:271–280, 1994.
- [FWB04] P. Felkel, R. Wegenkittl, and K. Buhler. Surface models of tube trees. In *Computer Graphics International, 2004. Proceedings*, pages 70 –77, 2004.
- [GPP⁺10] Olivier Gourmel, Anthony Pajot, Mathias Paulin, Loic Barthe, and Pierre Poulin. Fitted bvh for fast raytracing of metaballs. *Computer Graphics Forum*, 29(2):281–288, may 2010.
- [JLW10] Zhongping Ji, Ligang Liu, and Yigang Wang. B-mesh: A modeling system for base meshes of 3d articulated shapes. *Computer Graphics Forum*, 29(7):2169–2177, 2010.
- [LB06] Charles Loop and Jim Blinn. Real-time gpu rendering of piecewise algebraic surfaces. In *ACM SIGGRAPH 2006 Papers*, SIGGRAPH '06, pages 664–670, New York, NY, USA, 2006. ACM.
- [LC87] William E. Lorensen and Harvey E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. *SIGGRAPH Comput. Graph.*, 21:163–169, August 1987.
- [SE05] Jo Skjermo and Ole Christian Eidheim. Polygon mesh generation of branching structures. In Heikki Kalviainen, Jussi Parkkinen, and Arto Kaarna, editors, *Image Analysis*, volume 3540 of *Lecture Notes in Computer Science*, pages 109–122. Springer Berlin / Heidelberg, 2005. 10.1007/11499145_76.
- [She99] Andrei Sherstyuk. *Convolution Surfaces in Computer Graphics*. PhD thesis, Monash, Monash, Australia, Australia, 1999.
- [WGG98] Brian Wyvill, Eric Galin, and Andrew Guy. The Blob Tree, Warping, Blending and Boolean Operations in an Implicit Surface Modeling System. *Implicit Surfaces*, 3, June 1998. Chosen for inclusion in a special issue of Computer Graphics Forum.