Surface Subdivision

Zeming Liu

April 26, 2018

Objective

1.Read/create obj format models.
2.Implement Catmull Clark subdivision surface algorithm.

Extras

1.Implement Doo Sabin subdivision surface algorithm
2.Implement Loop subdivision surface algorithm
3.Construct and triangulate the models with bi-cubic bezier patches.
4.Compare these different schemes.
5.Make the schemes work on complicated models, and optimize the algorithms.

Read/Create .obj files

Creating an obj file from points,normals, and vertex textures are harder than reading information from them. The points,normals,and vertex textures are indexed naturally starting from one. and the faces(polygons) are represented by the indices of corresponding elements.

Since polygons share points and normals in a polygon meshes,this representation can save disk space largely. The subdivision schemes are applied recursively,so the vertices shrinking procedure is required.For most subdivision schemes,one needs to find the adjacent faces,edges for the vertices,and these can not be computed correctly if there exist duplicate points. I used to apply a naive algorithm for this procedure, and it's getting too slow when the model is getting larger, so I optimized it with a hash-based dictionary structure.

Catmull Clark

Algorithm

1.create an edge point for each edge
2.create a face point for each face
3.compute the new position for each vertex
4.generate new faces: a new face is generated by connecting the face point and any new vertex point and 2 adjacent edge points.

Properties

1.Catmull Clark takes a mesh of n-gons,which n can be any number and it does not need to be equal for every polygon in the mesh.
2.The subdivided mesh consists only tetragons(4 sides).

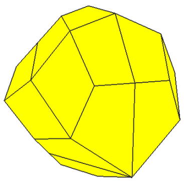

Figure 1: iteration 1

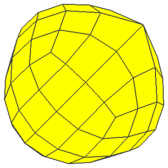

Figure 2: iteration 2

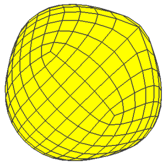

Figure 3: iteration 3

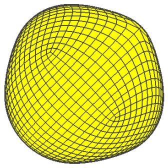

Figure 4: iteration 4

iteration	points	faces	file size	time(sec)
0	8	12	-	-
1	36	38	4KB	0.001468
2	144	146	12KB	0.005559
3	576	578	40KB	0.029669
4	2304	230	156KB	0.090214

Doo Sabin

Algorithm

1.create a face point which is the average of it's vertices for each face.
2.For each vertex of every original polygon, create the new point which is the average of the original vertex and the center point of the face.
3.Create new faces(3 types of faces):

1) F type faces: connecting each new vertices for all original faces,this polygon has the same number of edges as the old faces.
2) E type faces: for each edge,connect all new vertices which their old vertices share the edge,this type of polygons are all 4-gons.
3) V type faces: for each edge,connect all new vertices which shared the single old vertex,the number of edges of this type of faces can be any number.

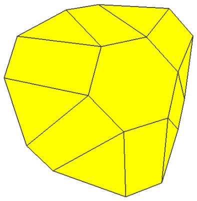

Figure 5: iteration 1

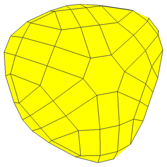

Figure 6: iteration 2

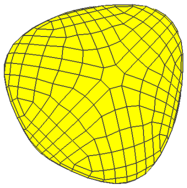

Figure 7: iteration 3

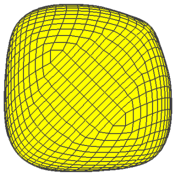

Figure 8: iteration 4

iteration	points	faces	file size	time(sec)
0	8	12	-	-
1	38	36	4KB	0.004573
2	146	144	8KB	0.176229
3	578	576	36KB	0.705481
4	2306	2304	144KB	0.282492

Loop subdivision

Algorithm

1.Computer Edge point as a combination of the end points of the edge and the other 2 points of the adjacent triangles for each edge(assuming the model consists only triangles).
2.Compute Vertex point based on all the connecting vertices and itself for each vertex.
3.Connect each new vertex and 2 adjacent edge points to form a new triangle.So,each triangle will generate 4 subdivided triangles.

Properties

1.Loop subdivision takes only triangular mesh.So a triangulation has to be done before run the subdivision.
2.It outputs triangular mesh with 4 times of the original number of triangles.

Triangulation

Loop subdivision takes only triangular meshes, so a triangulation process is required before applying the scheme (only required for the first iteration,since the output mesh is triangular).My triangulating scheme is to find the center of each polygon firstly,and triangulate each polygon by connecting the center to each vertex.

In the examples, I triangulate the polygons for all iterations,so the number of triangles are growing rapidly,as you can see in the figures below.

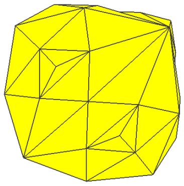

Figure 9: iteration 1

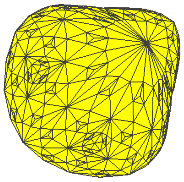

Figure 10: iteration 2

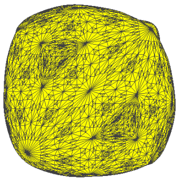

Figure 11: iteration 3

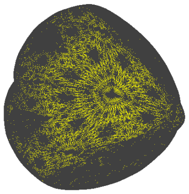

Figure 12: iteration 4

iteration	points	faces	file size	time(sec)
0	8	12	-	-
1	74	144	4KB	0.009150
2	866	1728	64KB	0.172394
3	10370	20736	820KB	1.649955
4	124414	248832	11MB	21.727128

bi-cubic bezier patches

I also tried to interpolate the meshes of the model with bi-cubic bezier patches,this is extended from one of the assignment program. The difference between interpolating bezier patches and the subdivision curves above is that the positions of new vertices of bi-cubic patches depend on the normals of original vertices. It will generate different shapes on the same original mesh given different normals.

The 2 subdivided models are generated with the same model as all models above.But the left one is generated from the cube whose normals are perpendicular to the faces(I used different normals for one point on different faces),while the right one is generated from the same model whose normals are pointing out from the center of the cube.

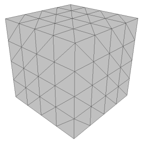

Figure 13: normals are perpendicular to the faces

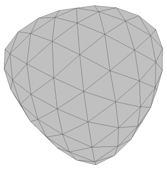

Figure 14: normals are pointing out from the center

More Models

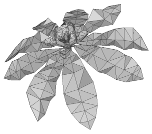

Figure 15: Model 1:original

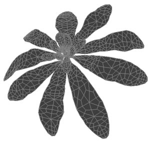

Figure 16: Model 1:Catmull-Clark

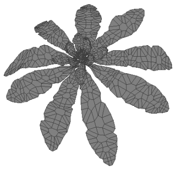

Figure 17: Model 1:Doo-Sabin

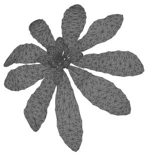

Figure 18: Model 1:Loop

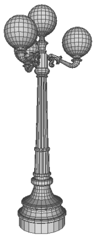

Figure 19: Model 2:original

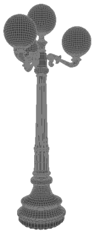

Figure 20: Model 2:Catmull-Clark

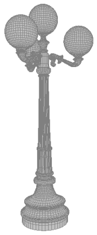

Figure 21: Model 3:Doo-Sabin

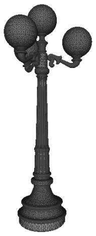

Figure 22: Model 3:Loop

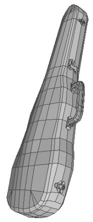

Figure 23: Model 3:original

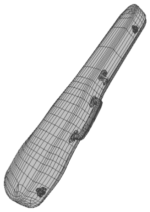

Figure 24: Model 3:Catmull-Clark

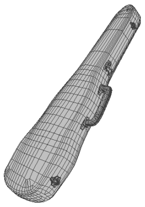

Figure 25: Model 3:Doo-Sabin

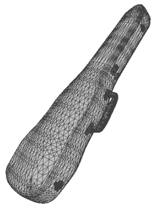

Figure 26: Model 3:Loop

What I have learned

1. obj file format, and some other 3d file formats.
2. implementation of several subdivision surface schemes.
3. Numerical problem is a big deal.
4. Things are much more complicated than what they appear.

References

[1] Edwin Earl Catmull,James Henry Clark. *Recursively generated B-spline surfaces on arbitrary topological meshes*

[2] D.W.H. Doo. *A Subdivision Algorithm For Smoothing Down Irregular Shaped Polyhedrons*

[3] Charles Teorell Loop. *Smooth Subdivision Surfaces Based on Triangles*

[4] Models used in this project:John Burkardt.'http://people.sc.fsu.edu/ jburkardt/data/obj/obj.html' *Department of Scientific Computing at Florida State University*

[5] MeshLab *http://www.meshlab.net/*