Tesselatior

Nidal Guerouaja

MAT 856092

New Features

- Cross platform build using CMake
- Dependency management with vcpkg
- Ui controls for lights, camera movement
- Automatic detection of mesh type on import
- Displacement map support
- Phong tessellation (https://perso.telecom-paristech.fr/boubek/papers/PhongTessellation/PhongTessellation.pdf)

Code rewrite

- Class Scene that holds IRenderableObjects
- The IRenderableObject can be:
  - Static Mesh
  - Subdiv Mesh
  - Terrain
  - Anything
- You create IRenderableObjects thru the Factory method (SubdivMeshCreator...)

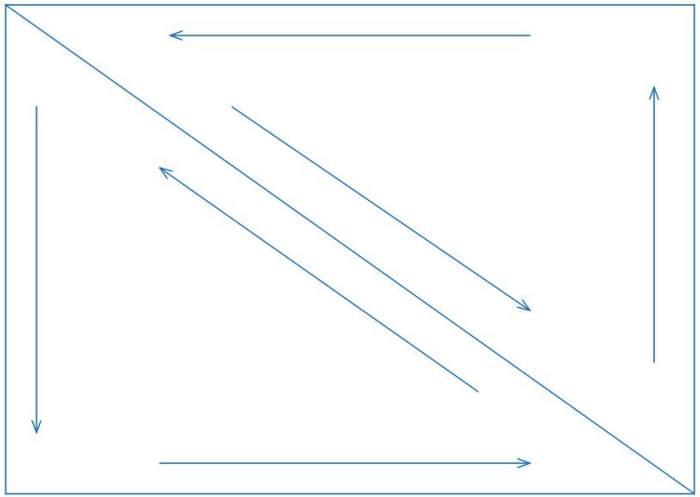
Code rewrite

Every IRenderableObjects holds a IMesh. It is polymorphic so it does not matter if it is a Triangle mesh or a Quads mesh

Every Mesh has a pointer to its halfedge data structure

The shaders are handled by a ShaderManager class (flyweight design pattern)

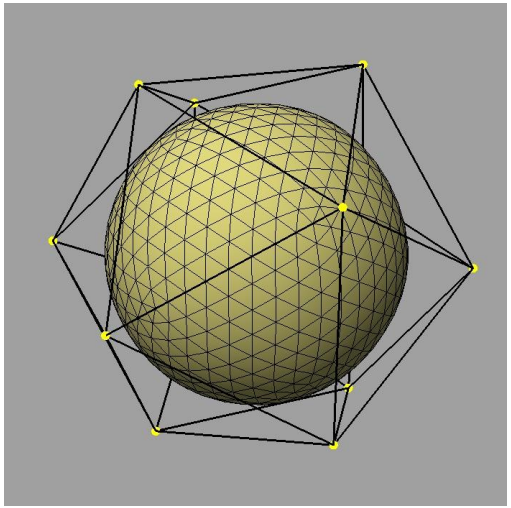
HalfEdge



```
struct Edge;
struct Face;
struct HalfEdge;

struct Vertex {
  glm::vec3 position;
  glm::vec3 normal;
  glm::vec2 text_coords;

  HalfEdge* halfedge;  // one of it's outgoing halfedge
};

struct HalfEdge {
  HalfEdge* next;
  HalfEdge* twin;  // opposite
  Vertex* vert;    // The vertex that the halfedge points to

  // this halfedge belongs to a face and an edge
  Face* face;
  Edge* edge;
};


struct Face {
  HalfEdge* halfedge;
};

struct Edge {
  HalfEdge* halfedge;
};
```

Subdivision surface

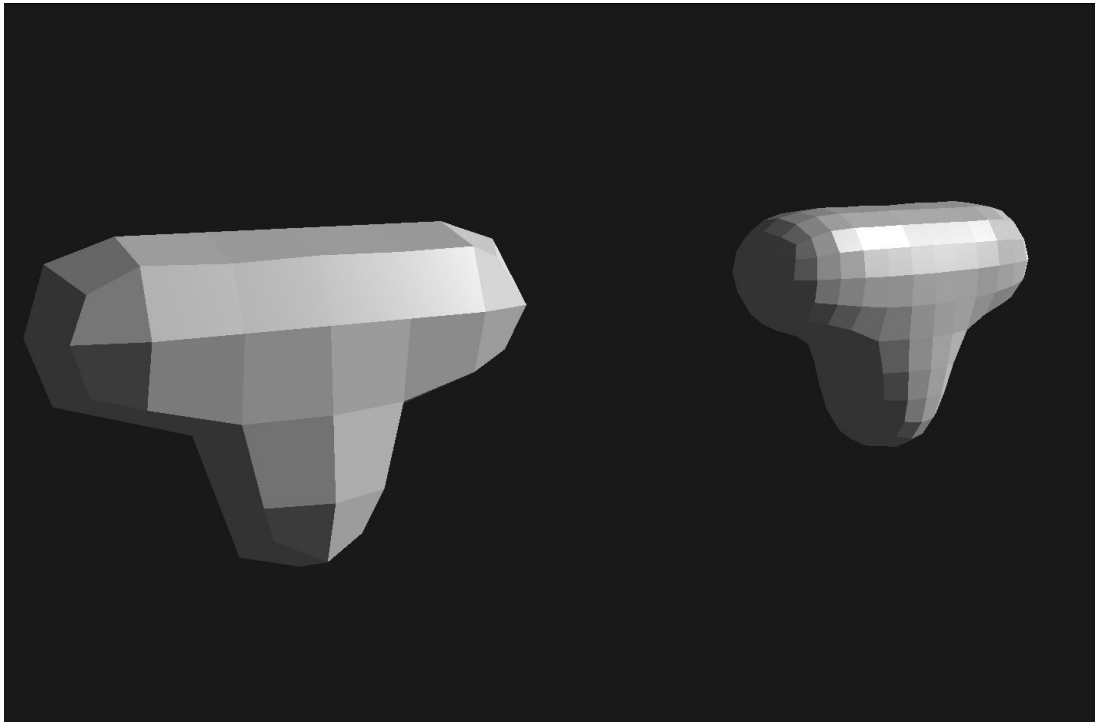
- Subdivision operates on a cage and produces a refined cage
- Tessellation operates on a surface and produces a discretization of that surface



Subdivision surface

Implementation:

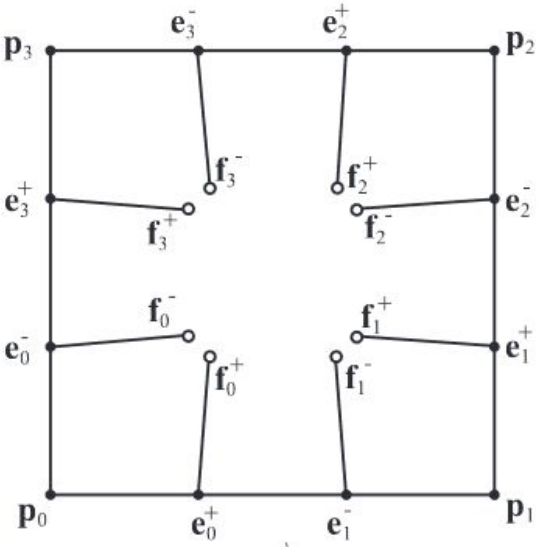
- Loop subdivision (with boundaries)
- Sqrt3 subdivision
- Catmull Clark subdivision



Gregory Patch (TODO)

- Build the patch
  - On the cpu
  - Using vertex shaders
  - Using tessellation control shaders
- Evaluate the patch on the Tessellation Evaluation Shader

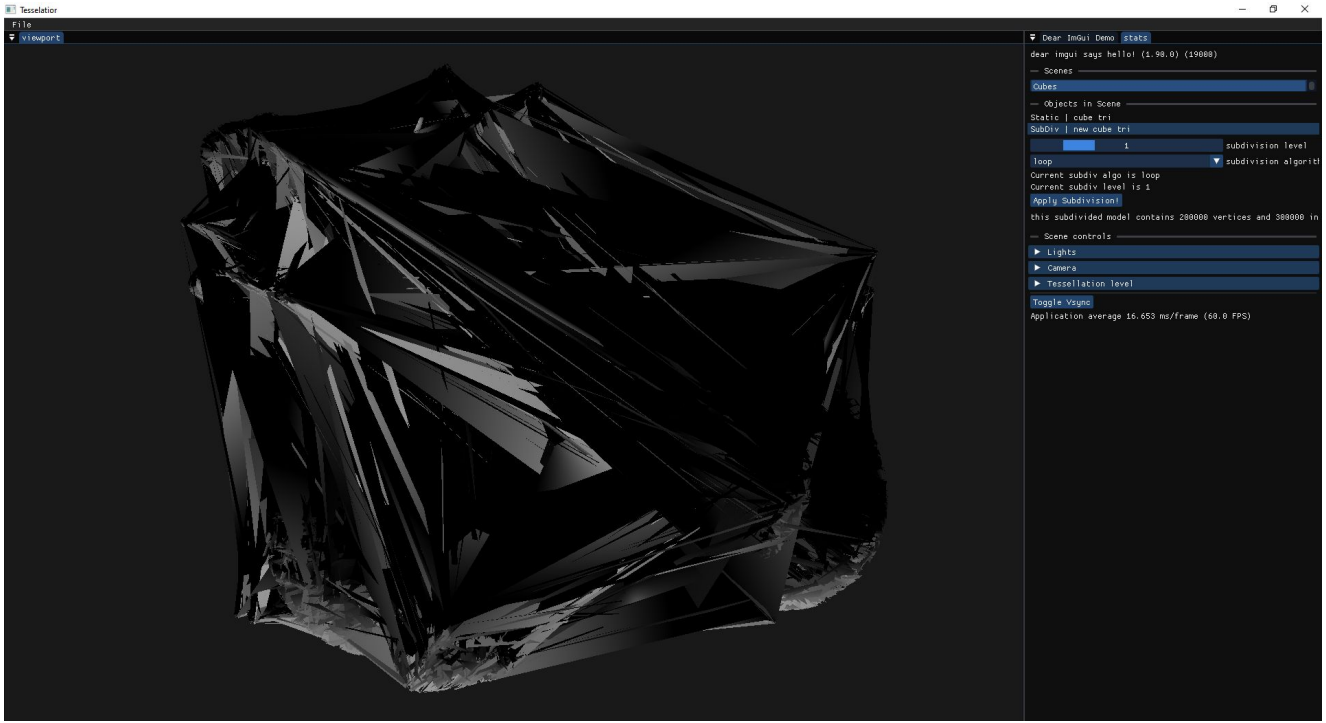
- https://people.engr.tamu.edu/schaefer/research/greg.pdf



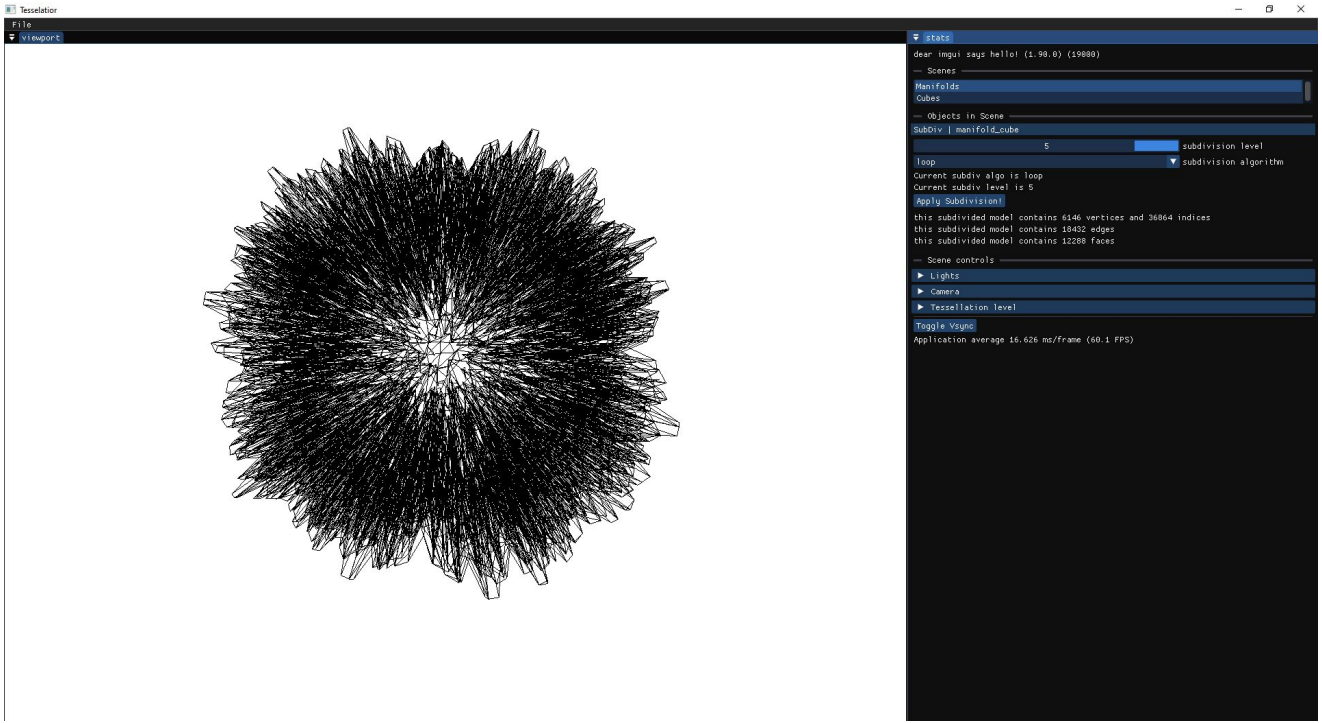
To do

- Improve architecture (remove code duplication, try different patterns / UI)
- Finish boundary cases, all subdivision algorithms
- Headless version, for testing performance and completeness
- Implementation of real time Subdivision Surface (using Gregory patches and tessellation shaders)
- ...

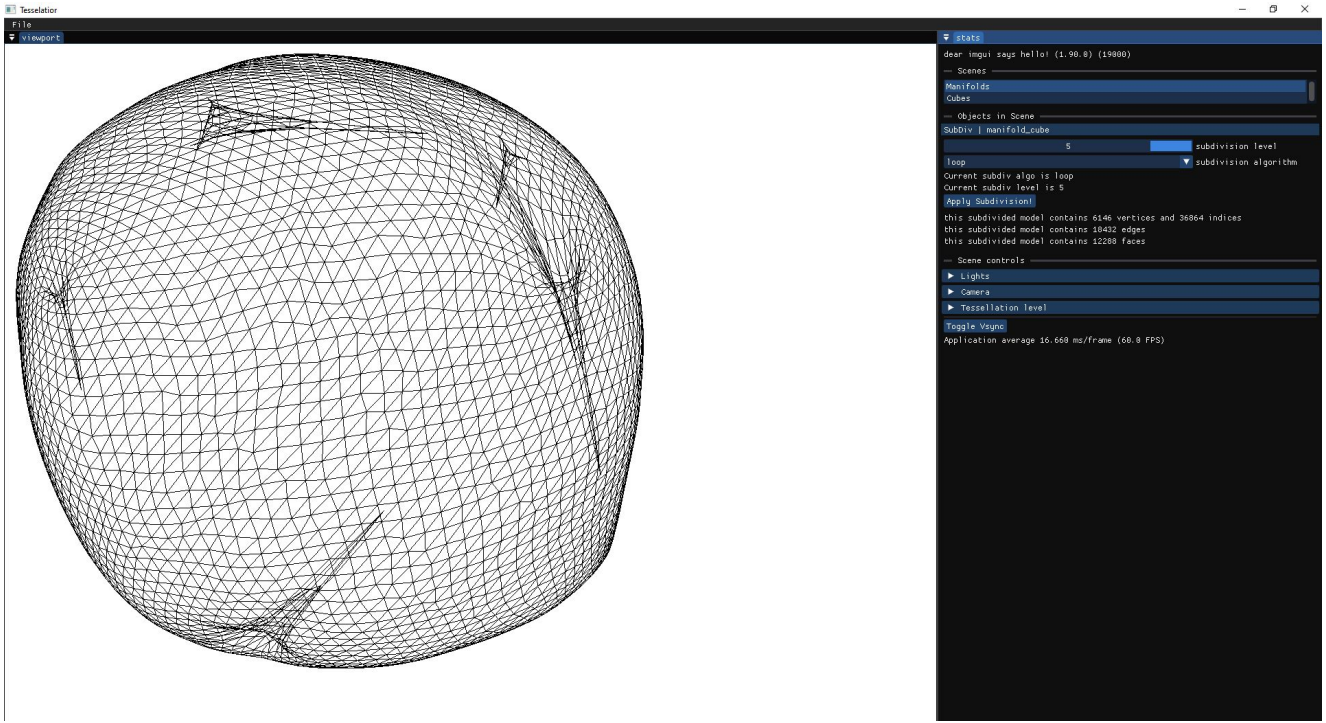
Bloopers


Tesselatior
File
viewport
Dear ImGui Demo
stats
dear imgui says hello! (1.90.0) (19080)
Scenes
Cubes
Objects in Scene
Static | cube tri
SubDiv | new cube tri
1
subdivision level
loop
subdivision algorith
Current subdiv algo is loop
Current subdiv level is 1
Apply Subdivision!
this subdivided model contains 200000 vertices and 380000 in
Scene controls
Lights
Camera
Tessellation level
Toggle Vsync
Application average 16.653 ms/frame (60.0 FPS)

Bloopers


Tesselatior
File
viewport
stats
dear imgui says hello! (1.90.0) (19080)
Scenes
Manifolds
Cubes
Objects in Scene
SubDiv | manifold_cube
5
subdivision level
loop
subdivision algorithm
Current subdiv algo is loop
Current subdiv level is 5
Apply Subdivision!
this subdivided model contains 6146 vertices and 36864 indices
this subdivided model contains 18432 edges
this subdivided model contains 12288 faces
Scene controls
Lights
Camera
Tessellation level
Toggle Vsync
Application average 16.626 ms/frame (60.1 FPS)

Bloopers


Tesselatior
File
viewport
stats
dear imgui says hello! (1.90.8) (19080)
Scenes
Manifolds
Cubes
Objects in Scene
SubDiv | manifold_cube
5
subdivision level
loop
subdivision algorithm
Current subdiv algo is loop
Current subdiv level is 5
Apply Subdivision!
this subdivided model contains 6146 vertices and 36864 indices
this subdivided model contains 18432 edges
this subdivided model contains 12288 faces
Scene controls
Lights
Camera
Tessellation level
Toggle Vsync
Application average 16.660 ms/frame (60.0 FPS)