Student : OLIVEIRA MACHADO DE SOUSA, Lucas
IMA208 - TP Reconstruction

CGAL's class Delaunay_triangulation_3 represents a three-dimensional Delaunay triangulation, which has the "empty sphere property": for each cell of this triangulation, its circumscribing sphere doesn't contain any other vertices of the triangulation in its interior.

To obtain an alpha-shape given a Delaunay triangulation, we have to go through each triangle of this structure and check if the radius of its circumscribing circle is smaller than α. We'll keep the vertices of the triangle only if it satisfies this requirement. In the end, from the set S of points we'll obtain a new set C, called alpha complex. The union of the triangles in the alpha complex gives us an alpha-shape.

From CGAL's documentation, we can see that the Delaunay triangulation gives us several tetrahedrons. Therefore, to verify every triangle in the triangulation, we'll perform a check on the four faces of each tetrahedron. Since the library only provides us with a function to compute squared distances, we'll be using the squared radius in comparison to α^2. The program can be executed as follows:

./Reconstruct <Image.xyz> <Alpha>
e.g. ./Reconstruct Bunny.xyz 0.001

We can see in the figures below the results for the Bunny figure.

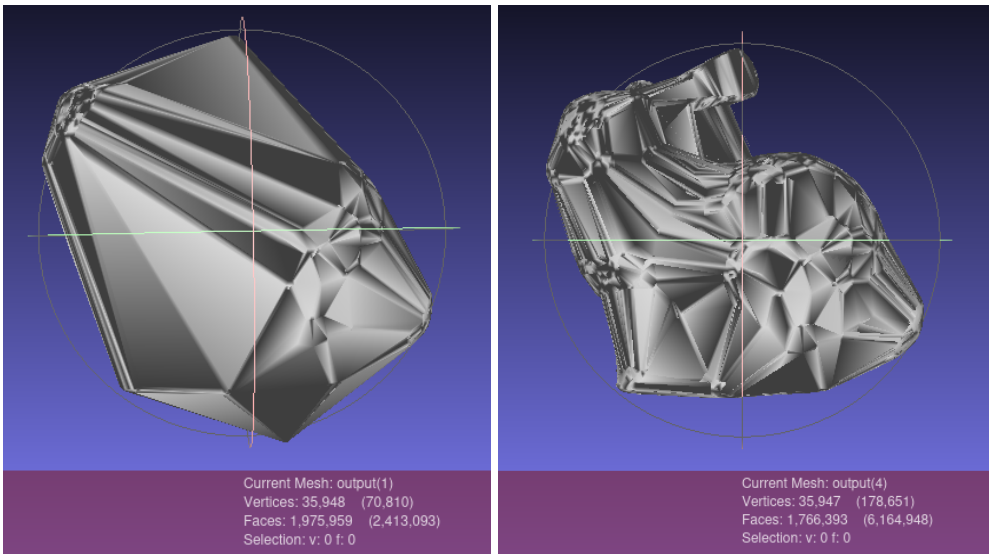

Figure 1 - Bunny figure for alpha = 1000 and alpha = 0.016.

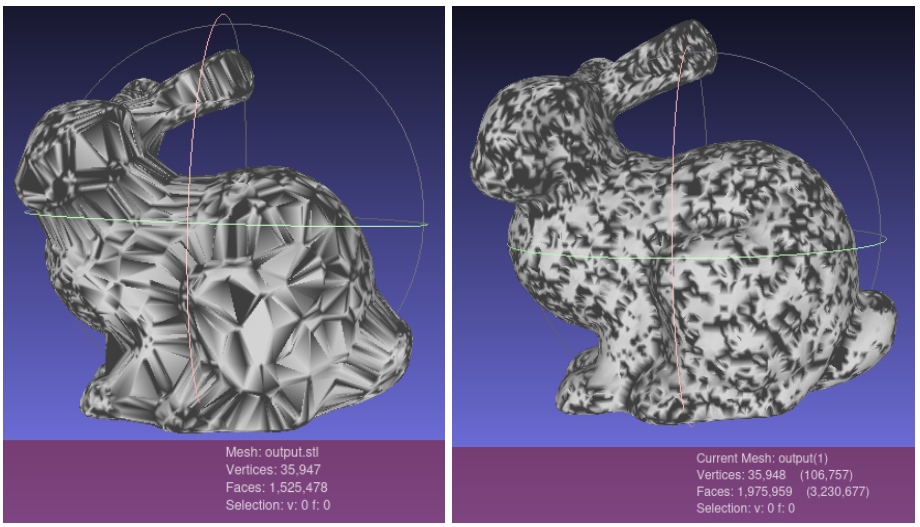

Figure 2 - Bunny figure for alpha = 0.008 and alpha = 0.002.

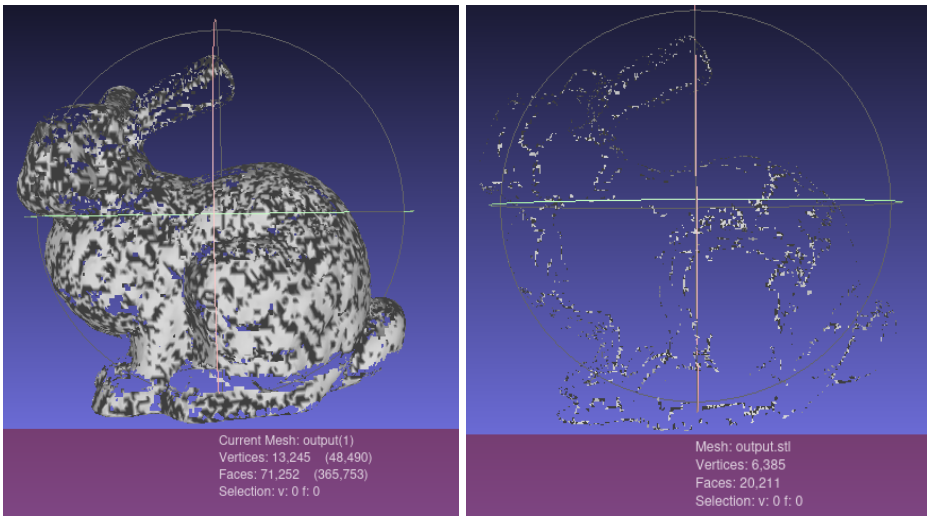

Figure 3 - Bunny figure for alpha = 0.0009 and alpha = 0.0008.

We can see from the images above that we get a good result for the bunny figure when the value of alpha is around 0.002. We see as well how sensitive the algorithm is when the value is lower than that. With alpha = 0.0009 we can see some parts of the bunny fading away, and with alpha = 0.0008 we can't see but a few points.

These results corroborate with the ones seen in class:

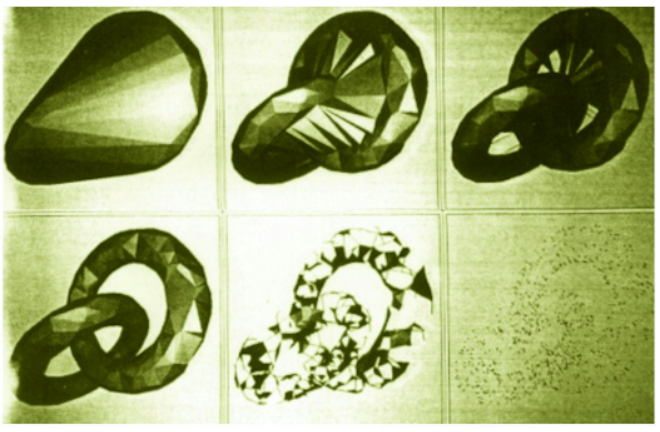

Figure 4 - Example seen in class, with alpha going from +∞ to 0.

The C++ code for this TP is sent with this report.
We can see below the results for the other figure provided, “Bimba.xyz”:

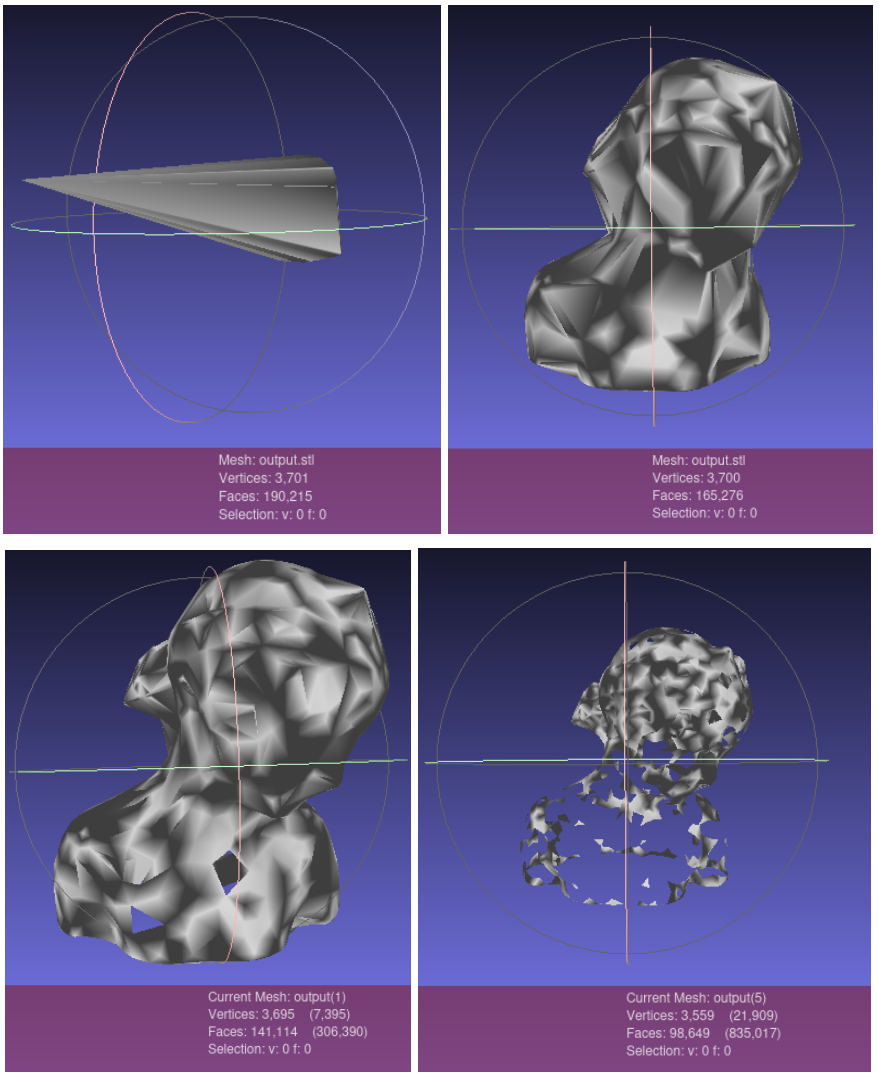

Figure 5 - Bimba.xyz with alpha = 1, 0.032, 0.016 and 0.008, respectively

The extra steps weren’t done due to the short deadline, but from this discussion in INRIA’s (Institut National de Recherche en Informatique et en Automatique) forum, “the ‘optimal’ alpha-value is the smallest alpha value that will give a single connected component including all data points.” This comment is a good starting point for a code aiming to find the best value for alpha.

INRIA’s Forum link: https://sympa.inria.fr/sympa/arc/cgal-discuss/2013-11/msg00110.html