Poisson Surface Reconstruction with Local Mesh Simplification

Jorge Henrique Piazentin Ono, Antônio Carlos Sementille, Marco Antônio Corbucci Caldeira, João Fernando Marar
Department of Computing, School of Sciences
Universidade Estadual Paulista - UNESP
Bauru, Brasil

***Abstract*—Surface reconstruction techniques have applications in many fields, such as cartography, augmented reality and reverse engineering. In real time systems, meshes in multiple resolutions can be used to ensure fast processing, where more detailed representations are used only when necessary. This paper proposes and validates the simplification of multiresolution meshes generated with Poisson Surface Reconstruction. We achieved simplifications of almost 90% of the reconstructed model maintaining it's average form.**

***Keywords*-Surface reconstruction; multiresolution; meshing; geometric modelling;**

I. Introduction

Virtual representations of objects can be created in CAD (Computer Aided Design) tools or generated from existent physical models, through surface reconstruction techniques. The goal of surface reconstruction is to find a surface from a finite set of sampled geometric values (usually points in space) [1]. The generated surface can be, then, discretized, producing a triangle mesh.

The simpler the mesh, the easier it is to store and process it. Multiresolution methods can be used to generate meshes in different level of detail (LOD) and, to ensure fast processing, an object which is far from the observer can be rendered in a low LOD, while higher LOD could be used for closer objects [2], [3]. Simplified meshes allow not only improvements in rendering speed, but also fast transmission of 3D models in network-based applications [4].

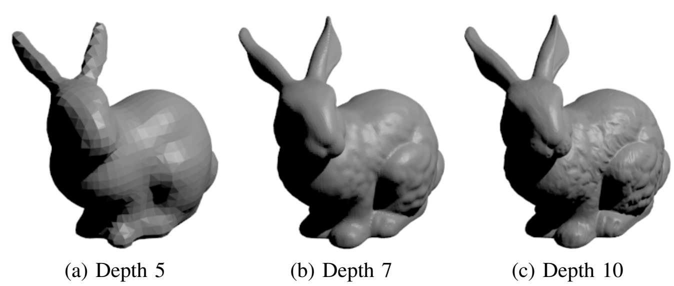
Contributions: This paper proposes an improvement to the Poisson surface reconstruction algorithm [5], by adding mesh simplification to the multiresolution result. Two approaches are studied: edge collapse operations [6] and a solution inspired by the work of Mingyi et al. [7], which defines a local normal change threshold to reconstruct a surface from the point set.

A. Related work

Surface reconstruction algorithms from scattered data points can be divided in 4 groups: distance functions, spacial subdivision, spatial free form warping and incremental surface-oriented construction [1]. Distance functions calculate the distance D between any point in space and the surface, therefore, when $D \approx 0$, the surface is approximated by an implicit function. Spatial subdivision methods decompose a set of points P in cells, selecting only the cells related to the form described by P. Spatial warping techniques deform an initial surface in order to approximate it to the point set. Incremental construction build an approximating/interpolating surface directly from properties of the data points [1].

The Poisson surface reconstruction [5] is an approach that expresses surface reconstruction as the solution to a Poisson equation. This method uses an implicit solution to approximate the surface and then extract the isosurface using an adaptation of the Marching Cubes [8] algorithm. Since accurate solutions are only needed near the surface, an octree structure is applied to approximate the raw data and, by choosing the maximum octree depth, one can create surfaces of various LOD. Fig. 1 shows our experiments with the Stanford bunny, which is reconstructed at 3 different octree depths. Table I shows the reduction of polygons as the maximum octree depth is limited.

Many methods can be used to simplify a mesh, for example, clustering algorithms, iterative simplification and particle simulation [9]. Mingyi et al. [7] evaluate the neighbourhood of every point and, if the local normal change, defined as the maximum angle between the points' normal, is less than a established threshold, these points are simplified by one plane. Edge collapse operations consist of iteratively replacing an edge with a single vertex by removing 2 triangles per collapse, but maintaining the overall look of the surface [10].



(a) Depth 5 (b) Depth 7 (c) Depth 10

Fig. 1. Stanford Bunny reconstructed with Poisson method at 3 different resolutions

II. Mesh simplification

Our technique aims at reducing the number of polygons produced by the Poisson method, by adding a surface simplification step to the reconstruction process. Two approaches are studied:

TABLE I
NUMBER OF TRIANGLES IN THE RECONSTRUCTED MODEL FOR DIFFERENT DEPTHS USING POISSON METHOD

Octree Depth	Number of vertex	Number of faces
5	2160	4316
7	36322	72640
10	682031	1364058

- Post-processing of the mesh, simplifying it with the edge collapse algorithm, well defined by [6], [11].
- Evaluation of the triangle mesh (local normal change of the faces), delimiting a threshold for coplanarity. This method is originally described by [7] to work on point sets, but can be extended for polygon meshes.

III. Implementation

In order to implement this multiresolution approach, a surface reconstruction pipeline was developed, consisting of data (point cloud) acquisition with Microsoft Kinect, point cloud filtering and segmentation, surface reconstruction and surface simplification.

The Point Cloud Library (PCL) [12] was used to capture and pre-process the raw data. Then we adapted the algorithm of [5] to analyse and reconstruct the surface. Finally, we adjusted the CGAL [10] implementation of the edge collapse algorithm to simplify our triangle mesh.

IV. Work in progress

Currently, we are working on mesh segmentation of our point sets (depth maps) using the PCL [12] version of RANSAC (Random Sample Consensus) [13] and we are developing the local normal change approach for triangle mesh simplification.

V. Experiments

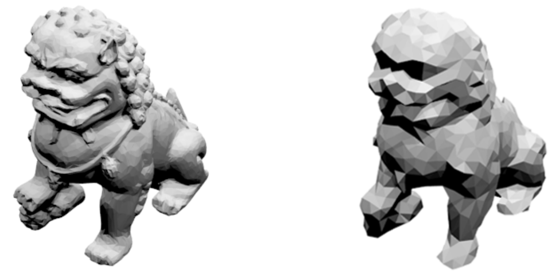
We applied our algorithm in public domain point sets (the Stanford 3D Scanning Repository and the CGAL Surface Reconstruction Examples). Fig. 2 shows a dragon reconstructed with Poisson method with octree at depth 8 and it's simplified version with Edge Collapse operation. Table II shows that even though the simplification process reduced the number of polygons to a tenth of the original model, the average form of the object is maintained.

TABLE II
SIMPLIFICATION RESULTS USING EDGE COLLAPSE ALGORITHM

	Number of vertex	Number of faces
Original Dragon	10000	19994
Simplified Dragon	1001	1996

VI. Conclusion and Future Work

In this paper, we proposed and validated the simplification of multi-resolution objects created with Poisson surface reconstruction. As we finish the implementation of our prototype, we expect to achieve better multi-resolution meshes (less polygons and more detail) with the combination of these techniques than with the Poisson method itself. In order to validate our experiments, we will use Metro [14] to compare the differences between the original and the simplified surface.



(a) Original dragon (b) Simplified dragon

Fig. 2. Edge Collapse Simplification

Acknowledgment

The authors would like to thank the use of the laboratory SACI (Adaptive Systems and Intelligent Computing), Department of Computing, School of Sciences, UNESP Bauru.

References

[1] H. Muller, "Surface reconstruction - an introduction," in *Scientific Visualization Conference, 1997*, Jun. 1997, p. 239.
[2] T. Gerstner, "Multiresolution extraction and rendering of transparent isosurfaces," *Computers & Graphics*, vol. 26, no. 2, p. 219–228, 2002.
[3] M. Botsch and L. Kobbelt, "A robust procedure to eliminate degenerate faces from triangle meshes," in *Proc. of Vision, Modeling, and Visualization*, vol. 1, 2001, p. 283–289.
[4] P. Cignoni, C. Montani, and R. Scopigno, "A comparison of mesh simplification algorithms," *Computers & Graphics*, vol. 22, no. 1, p. 37–54, 1998.
[5] M. Kazhdan, M. Bolitho, and H. Hoppe, "Poisson surface reconstruction," in *Proceedings of the fourth Eurographics symposium on Geometry processing*, 2006, p. 61–70.
[6] P. Lindstrom and G. Turk, "Fast and memory efficient polygonal simplification," in *Visualization'98. Proceedings*, 1998, p. 279–286.
[7] H. Mingyi, X. Bangshu, and Y. Huajing, "Multi-resolution surface reconstruction," in *Image Processing, 2004. ICIP'04. 2004 International Conference on*, vol. 3, 2004, p. 1971–1974.
[8] H. E. C. William E. Lorensen, "Marching cubes: A high resolution 3d surface construction algorithm," *SIGGRAPH '87, ACM*, vol. ACM-0-89791-227-6/87/007/01, pp. 163–169, 1987.
[9] M. Pauly, M. Gross, and L. Kobbelt, "Efficient simplification of point-sampled surfaces," in *Visualization, 2002. VIS 2002. IEEE*, 2002, p. 163–170.
[10] T. C. Project, *CGAL User and Reference Manual*, 4th ed. CGAL Editorial Board, 2012, http\string://www.cgal.org/Manual/4.0/doc_html/cgal_manual/packages.html.
[11] H. Hoppe, "Progressive meshes," in *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, 1996, p. 99–108.
[12] S. Rusu, R.B. ,Cousins, "3d is here: Point cloud library (pcl)," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, 2011, p. 1–4.
[13] M. Fischler and R. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, p. 381–395, 1981.
[14] P. Cignoni, C. Rocchini, and R. Scopigno, "Metro: measuring error on simplified surfaces," in *Computer Graphics Forum*, vol. 17, 1998, p. 167–174.