Surface Reconstruction from Point Set using Projection Operator

Ly Phan, Lu Liu, Sasakthi Abeysinghe, Tao Ju, Cindy M. Grimm
Washington University in St. Louis *

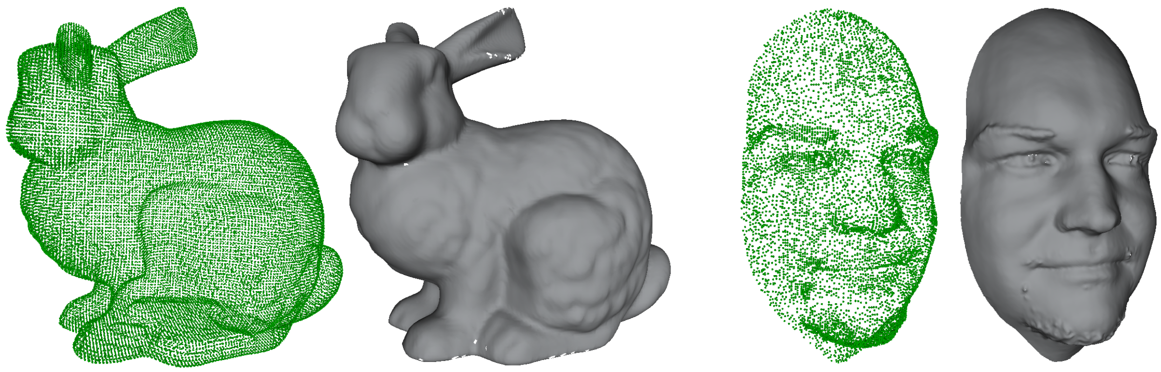

Figure 1: (a) Input point set for the Stanford bunny, (b) output surface. (c) Input point set for a face, (d) output surface with boundary.

1 Introduction

Surface reconstruction from point clouds is an important problem for its wide range of practical applications. One way to describe a surface implied by point sets is to define a method that projects a given point in space onto the surface. In the seminal work of Amenta [Amenta and Kil 2004], a mathematical definition of such point set surfaces was proposed as the extremal surface of an unoriented vector field and an energy function. Although precisely defined, the surface was constructed indirectly by a projection process that results in a dense point set instead of explicit mesh geometry.

In this work, we propose a grid-based algorithm that directly extracts the extremal surface geometry, given a smooth vector field and energy function. The key observation that enables this direct construction is that the extremal surface can be considered as the singularity of an oriented vector field, which can be computed directly using a contour-like approach. Using the new method, we compare and discuss our key findings about two different vector fields. We also propose a combination of vector fields that will allow our algorithm to generate surfaces with boundaries without spurious components exhibited by previous approaches.

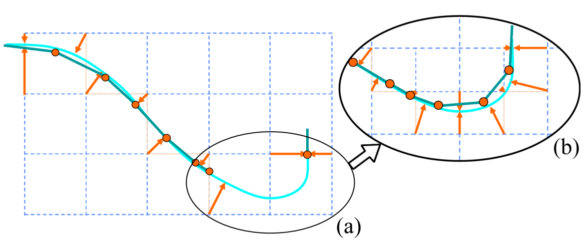

Figure 2: Our algorithm in a nutshell: We approximate the extremal surface (blue curve) and construct a simplicial surface (green curve) by examining the orientation and magnitude of the vectors (orange arrows) defined at two end points of each grid edge.

*e-mail: {faanly, ll10, ssa1, taoju, cmg}@cs.wustl.edu

2 Our Approach

Given a point set in space, we first construct a grid enclosing the point set. Using our choice of vector field, we calculate a vector at each grid point directed toward the surface. For each edge of the grid, if the vectors at its two ends are in opposite directions, we mark it. By minimizing the energy function along the vectors, we can calculate the distance from each end point to the surface. Using that information, we approximate the intersection point of the edge and the surface. For each cell with marked edges, we compute a cell centroid as the average of intersection points on its marked edges. Finally, for each marked edge, we construct a face with the centroids of its incident cells as vertices. Since we do not construct a global inside-outside function, our method can handle surfaces with boundaries given a suitable vector field.

For our experiments, we use two vector fields: gradient field or local neighborhood vector field. Our first key finding is that the local neighborhood field is much more efficient in terms of runtime. However, since the gradient field is calculated from the distance field, it can handle surfaces with boundary, while the local neighborhood approach cannot. In our future work, we plan to combine these two vector fields so that we can handle surfaces with boundary efficiently.

Our second key finding is that if the resolution of our grid is not fine enough at places where the surface has high curvature, there will be holes on the surface reconstructed (as can be seen on the ears, neck and feet of the bunny in Figure 1b). This is because the vector field in the vicinity changes very fast due to the high curvature (Figure 2a). An adaptive grid, such as one implemented using an octree, will be able to resolve this problem without compromising speed (Figure 2b).

References

AMENTA, N., AND KIL, Y. J. 2004. Defining point-set surfaces. *ACM Transactions on Graphics (SIGGRAPH 2004 Proceedings) 23*, 3, 264–270.