Non-Convex Hull Surfaces

Gabriel Taubin* Brown University



Figure 1: A: A 2D oriented point cloud. B: A supporting linear half space for one of the oriented points. C: The oriented convex hull (OCH) of the point cloud. D: An outside supporting circle for one of the points. E: Inside supporting circles are obtained by inverting the orientation vectors. F: The non-convex hull (NCH) of the oriented point cloud is the intersection of the complement of all the outside supporting circles. G: A 3D oriented point cloud. NCH surface reconstructions based on octrees of depth 7 (H), 8 (I), and 9 (J).

Abstract

We present a new algorithm to reconstruct approximating watertight surfaces from finite oriented point clouds. The Convex Hull (CH) of an arbitrary set of points, constructed as the intersection of all the supporting linear half spaces, is a piecewise linear watertight surface, but usually a poor approximation of the sampled surface. We introduce the Non-Convex Hull (NCH) of an oriented point cloud as the intersection of complementary supporting spherical half spaces; one per point. The boundary surface of this set is a piecewise quadratic interpolating surface, which can also be described as the zero level set of the NCH Signed Distance function. We evaluate the NCH Signed Distance function on the vertices of a volumetric mesh, regular or adaptive, and generate an approximating polygonal mesh for the NCH Surface using an isosurface algorithm. Despite its simplicity, this simple algorithm produces high quality polygon meshes competitive with those generated by state-of-the-art algorithms. The relation to the Medial Axis Transform is described.

CR Categories: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Curve, surface, solid, and object representations;

Keywords: surface reconstruction, oriented point clouds, convexity

1 Introduction

In this paper we are concerned with the problem of reconstructing an oriented watertight surface approximating a finite set of points with associated unit length orientation vectors, consistently oriented with respect to the sampled surface S. Oriented point clouds are produced by laser scanners, structured lighting systems, multiview stereo algorithms, and simulation algorithms.

The prior art on surface reconstruction from point clouds is extensive, spanning more than two decades. Most combinatorial algorithms produce interpolating polygon meshes, and some come with guaranteed reconstruction quality [Bernardini et al. 1999; Amenta et al. 2001; Dey 2007].

Since implicit surfaces are watertight, most approximating surface reconstruction methods produce implicit surfaces, and through variational formulations reduce the problem to the solution of large sparse linear systems [Kazhdan et al. 2006; Alliez et al. 2007; Manson et al. 2008; Hoppe et al. 1992; Boissonnat and Cazals 2002; Calakli and Taubin 2011; Alexa et al. 2003; Fleishman et al. 2005; Ohtake et al. 2005]. The estimated implicit function is often evaluated on a regular grid of sufficient resolution, and a polygon mesh approximation is generated using an isosurface algorithm such as Marching Cubes [Lorensen and Cline 1987]. Since the cost of estimating or evaluating the implicit function on a regular grid is often excessive, some methods perform the computations on adaptive volumetric meshes such as octrees which require more complex contouring algorithms [Ohtake et al. 2005; Kazhdan et al. 2006; Manson et al. 2008; Calakli and Taubin 2011]

The method proposed in this paper falls somewhere in between these categories. A continuous interpolating piecewise quadratic NCH Signed Distance function is constructed as a function of the oriented point locations and orientation vectors through a simple and direct computation. Then the NCH Signed Distance function is evaluated on the vertices of a volumetric mesh, such as a regular voxel grid or octree constructed as a function of the point locations. Finally an isosurface algorithm is also used to generate an approximating polygonal mesh. When the volumetric mesh is conforming, the polygon mesh is guaranteed to be watertight.

^{*}e-mail: taubin@brown.edu



Figure 2: A: An oriented point cloud with approximately 25K points. B: The polygon mesh extracted by the naive algorithm on a 500^3 voxel grid. C: The oriented points superimposed with the mesh. D: Detail view of the point cloud showing points and orientation vectors. E: Close-up view of B. Close-up view of C. Note that the polygon density is higher than the point cloud sampling rate. The reconstructed polygon mesh has 556,668 vertices and 555,386 faces.

2 The Naïve Algorithm

To emphasize the simplicity of the proposed method, in this section we describe what we call the Naïve NCH Surface Reconstruction algorithm. In subsequent sections explain why it works, and variations. The Naïve NCH Surface Reconstruction algorithm for a finite set of oriented points comprises three steps: 1) estimating one NCH Signed Distance parameter for each data point; 2) evaluating the NCH Signed Distance function on the vertices of a volumetric mesh such as a regular voxel grid or octree; 3) approximating the zero level set of the NCH Signed Distance function by a polygon mesh using an isosurface algorithm such as Marching Cubes [Lorensen and Cline 1987].

For a finite set of points $\mathcal{P} = \{p_1, \ldots, p_N\}$, with associated unit length orientation vectors n_1, \ldots, n_N we define the value of the NCH Signed Distance function at a 3D point x as the maximum over N basis functions,

$$f(x) = \max_{1 \le i \le N} f_i(x) \tag{1}$$

where for each oriented point (p_i, n_i) , we have one basis function

$$f_i(x) = n_i^t(x - p_i) - \rho_i ||x - p_i||^2$$
(2)

The parameter ρ_i is set equal to zero if the set J_i of indices j = 1, ..., N such that $n_i^t(p_j - p_i) > 0$ is empty, and otherwise

$$\rho_i = \min\left\{\frac{n_i^t(p_j - p_i)}{\|p_j - p_i\|^2} : j \in J_i\right\} > 0.$$
(3)

Figure 2 shows one result obtained with exactly this algorithm.

The main advantage of the Naïve NCH Surface Reconstruction algorithm is its simplicity, since it can be implemented literally with only a few lines of code. The main disadvantage of the method is that its complexity is quadratic in the number of points. We address this issue through an adaptive subsampling approach which yields NCH Surfaces which interpolate only a subset of the input



Figure 3: The geometry of the spherical support functions $f_p(x)$.

points, and approximates the remaining points under user-specified maximum error.

Even though large areas of missing data points and holes are filled because the output mesh is watertight (except for its intersection with boundaries of the bounding box), the algorithm not always fills holes in an intuitive manner, as can be observed in the teaser figure. Some of the surface reconstruction algorithms based on variational principles mentioned in the introduction tend to fill holes in a more intuitive and predictable fashion. However, since it is not iterative, this algorithm is robust, and in many cases it can deal gracefully with uneven sampling, as shown in figure 5.

3 Non-Convex Hull

In this paper we refer to a half space as a set $H = \{x : f(x) \le 0\}$ of points satisfying an inequality constraint for a continuous realvalued function f(x) defined for every point x in a certain domain U contained in the ambient space (2D or 3D here). A linear half space is defined by a linear function f(x). Given a set of points \mathcal{P} , finite or infinite, and not necessarily oriented, the half space Hdefined above (with f(x) linear or non-linear) is said to be a supporting half space for \mathcal{P} if the following two conditions are satisfied: 1) the set \mathcal{P} is contained in H, i.e., $\mathcal{P} \subseteq H$; and 2) there is at least one point p in \mathcal{P} where the function attains the value zero f(p) = 0. The Convex Hull $CH(\mathcal{P})$ of the set \mathcal{P} can be defined as the intersection of all the supporting linear half spaces for \mathcal{P} . Since a linear half space is a convex set, and convexity is preserved by intersection, $CH(\mathcal{P})$ is also a convex set.

For the finite set of points $\{p_1, \ldots, p_N\}$, with associated unit length orientation vectors n_1, \ldots, n_N we adopt a more restricted definition which takes into account the point orientations. Each oriented point p_i defines a linear half space $H_i = \{x : f_i(x) \le 0\}$ for the linear function $f_i(x) = n_i^t(x - p_i)$. We define the Oriented Convex Hull of the set of points OCH(\mathcal{P}) as the intersection of all the supporting linear half spaces H_i . Note that if the orientation of the points is reversed, a completely different result is usually obtained, since the family of supporting linear half spaces is different, and often even empty.

Since the Oriented Convex Hull is a convex set, it cannot approximate the boundary surface of an object with concavities. To be able to approximate these surfaces we need to augment the family of supporting half spaces. It is necessary for this family to include non-convex half spaces, so that their intersection can represent solid objects with concavities. For each point p_i in the data set \mathcal{P} with associated orientation vector n_i , and every positive value of r > 0, we consider the function

$$f_i^r(x) = \frac{1}{2r} \left\{ r^2 - \|x - (p_i + r \, n_i)\|^2 \right\}$$
(4)

defined for x in the domain U. This function is positive inside a sphere of radius r centered at the point $q = p_i + r n_i$, negative outside the sphere, attains its maximum value r/2 at the center point



Figure 4: *Results on evenly sampled low noise surfaces. Left: Oriented points. Center: Reconstruction with an octree of depth 9. Right: Reconstruction with an octree of depth 10.*

q, and has unit slope $\|\nabla f_i^r(x)\| = 1$ at every point x where $f_i^r(x)$ is equal to zero.

As shown in figure 3, the point $q = p_i + r n_i$ is located on the ray defined by the point p_i and vector n_i at distance r from p_i . In addition, $f_i^r(p) = 0$, and $\nabla f_i^r(p) = n_i$ for all values of r. Now we define r_i as the largest value of r so that $f_i^r(p_j) \leq 0$ for every other point $p_j \in \mathcal{P}$. For finite sets of oriented points we have $r_i > 0$ for each data point $p_i \in \mathcal{P}$. As a result, the half space H_i defined by the function $f_i(x) = f_i^{r_i}(x)$ of equation 2 is supporting, where $\rho_i = 1/(2r_i) > 0$. We refer to these sets H_i as complementary spherical supporting half spaces. For the linear supporting half spaces to be included as special cases, we need to allow $\rho_i = 0$, or $r_i = \infty$. Note that, as opposed to the Oriented Convex Hull case, here every oriented point p_i in the data set has an associated supporting half space H_i , and if the orientations are reversed $(n_i \mapsto -n_i)$, completely different half spaces are obtained.

The value of ρ_i for an oriented point p_i is computed as the minimum over all the positive values

$$\rho_{ij} = \frac{n_i^t (p_j - p_i)}{\|p_j - p_i\|^2} \tag{5}$$

over all $j \neq i$. Note that if $\rho_{ij} \leq 0$ for all $j \neq i$, then we should set $\rho_i = 0$, because in this case the linear half space $H_i = \{x : n_i^t(x - p_i)\}$ is supporting for the set \mathcal{P} .

We define the Non-Convex Hull of the oriented point set \mathcal{P} , denoted NCH(\mathcal{P}), as the intersection of all the complementary spherical supporting half spaces H_i , as defined above. This is the same as saying that the complement of NCH(\mathcal{P}) is a union of balls. Note that NCH(\mathcal{P}) is also a half space. Namely, the half space defined by the continuous signed distance function f(x) shown in equation 1. This function is well defined when the data set \mathcal{P} is bounded, and in particular when it is finite.



Figure 5: *Results on unevenly sampled surfaces. Left: Oriented points. Center: Reconstruction with an octree of depth 9. Right: Reconstruction with an octree of depth 10.*

4 Relation to the Medial Axis Transform

In this section we assume that the set of points \mathcal{P} is a sampling of the boundary surface S of a bounded solid object O, and that the object O is an open set in 3D. As a result, the surface S is bounded, orientable (separates the inside from the outside of O), closed, and it has no boundary (i.e., no holes). Furthermore we assume that it is smooth, has a continuous unit length normal field pointing towards the inside of O, and has continuous curvatures.

The Medial Axis Transform (MAT) is a representation of the object O as a union of balls. A ball $B = B(q, r) = \{x : ||x - q|| < r\}$ is an open sphere with a center q and a radius r > 0. Being an open set, the solid object O is equal to the union of all the balls Bcontained in O. But this representation is too redundant to be used in a practical surface reconstruction algorithm. Since the set of all balls contained in O is partially ordered by inclusion, the Medial Axis Transform of O can be defined as the family MAT(O) of balls B contained in O which are maximal with respect to inclusion. The balls that belong to the MAT(O) are called medial balls. Obviously, the solid object O is also equal to the union of all the medial balls. This definition differs from the one given for example in [Amenta et al. 2001], which also includes the maximal balls contained in the outside of the object (complement of $S \cup O$), but this definition is more appropriate for our purposes. To be more precise we can refer to the Inside Medial Axis Transform, the Outside Medial Axis Transform, and the Symmetric Medial Axis Transform whenever necessary.

We define the Medial Axis MA(O) of O as the set of centers of medial balls. Since two different medial balls cannot have the same center, the mapping $MA(O) \rightarrow MAT(O)$ which assigns each medial ball center to the corresponding medial ball is well defined, 1-1 and onto. Another way of describing the Medial Axis Transform is as a set of points called Medial Axis Set, augmented with a nonnegative radius function which assigns to each medial ball center the corresponding medial ball radius. These radii are, of course, not independent of each other. In this paper we present an alternative description of the Medial Axis Transform, where each medial ball is not described by its center and radius, but by one of its boundary points and the radius. If B is a medial ball of center q and radius r, p is a point in the intersection of the boundary of B and S, and n_p is the surface normal to S at p pointing towards the interior of O, since the boundary of B and S are tangent, the ball center q must lie on the normal ray defined by p and n, in which case we have $q = p + rn_p$. In addition, because of the maximality of the ball B, the radius r is uniquely determined: it must be equal to the maximum of radii r' > 0 of balls centered at points q' = p + r'n lying on the ray defined by the point p and the vector n, fully contained in O. On the other hand, if p is a point on the surface S, since O is the union of all the medial balls, and S is the boundary of this set, a medial ball B must exists so that p belongs to the intersection of the boundary of B and S.

In summary, every medial ball can be written as $B(p + r_p n_p, r_p)$ for at least one surface point p, where

$$r_p = \max\{r' > 0 : B(p + r'n_p, r') \subset O\}.$$
 (6)

Note that in the mapping $S \to MAT(O)$ so defined, two or more surface points may map onto the same medial ball. But this is not a problem, and in fact it is an unusual event; what is important is that every medial ball is accounted for, so that the surface S can be reconstructed as the boundary of the union of balls

$$O = \bigcup \{ B(p + r_p n_p, r_p) : p \in S \} .$$
(7)

The surface S can is approximated as the boundary of the Non-Convex Hull NCH(\mathcal{P}) defined as a half space of the NCH Signed Distance f(x). The Local Feature Size LFS(p) at a surface point $p \in S$ is usually defined as the distance from p to the nearest point in the Symmetric Medial Axis [Amenta et al. 2001; Dey 2007]. A finite set $\mathcal{P} \subset S$ is defined as an ϵ -sample of S if the distance from any point $p \in S$ to its closest sample in \mathcal{P} is at most ϵ LFS(p). Several authors have shown that for sufficiently small ϵ the surface reconstructed as the boundary of MAT(\mathcal{P}) is a geometrically accurate approximation of S [Amenta et al. 2001; Dey 2007], and our experiments validate these theoretical results.

5 Octrees and Isosurfaces

Since typically the NCH Signed Distance function has constant sign in large regions, one way to potentially reduce the computational cost of the naïve algorithm is to detect those areas and not to evaluate the function there. Following a standard recursive space partition approach, we build an octree as a function of the point locations and the orientation vectors, we evaluate the NCH Signed Distance function at the vertices of the volumetric mesh, and use the Dual Marching Cubes (MC) algorithm [Schaefer and Warren 2005] to generate an output polygon mesh. Since the dual mesh of an octree is a conforming volumetric polyhedral mesh, the polygon meshes produced by DMC are adaptive, but have no cracks. The results shown in figures 4 and 5 have been computed using our implementation of DMC. Although the results presented are very good, we regard them as preliminary work. Due to lack of space, the details of this process as well as extensive experimental results will be presented in a future extended publication.

6 Conclusion

We have introduced an extremely simple algorithm to reconstruct watertight surfaces from finite sets of oriented points. The formulation generalizes the Convex Hull in such a way that concavities can be represented and approximated. We have also proposed preliminary strategies to reduce the computational cost by generating adaptive polygon meshes and by subsampling. Despite its simplicity, the proposed algorithm produces high quality polygon meshes competitive with those produced by state-of-the-art algorithms. Since the algorithm is massively paralellizable, and we plan to produce a GPU implementation in the near future.

This material is based upon work supported by the National Science Foundation under grants CCF-0729126, IIS-0808718, CCF-0915661, and IIP-1215308.

References

- ALEXA, M., BEHR, J., COHEN-OR, D., FLEISHMAN, S., LEVIN, D., AND SILVA, C. 2003. Computing and rendering point set surfaces. *IEEE Transactions on Visualization and Computer Graphics*, 3–15.
- ALLIEZ, P., COHEN-STEINER, D., TONG, Y., AND DESBRUN, M. 2007. Voronoi-based variational reconstruction of unoriented point sets. In *Proceedings of the fifth Eurographics symposium* on Geometry processing, Eurographics Association, 39–48.
- AMENTA, N., CHOI, S., AND KOLLURI, R. 2001. The Power Crust, Unions of Balls, and the Medial Axis Transform. *Computational Geometry Theory and Applications* 19, 2-3 (jul), 127– 153.
- BERNARDINI, F., MITTLEMAN, J., RUSHMEIER, H., SILVA, C., AND TAUBIN, G. 1999. The Ball-Pivoting Algorithm for Surface Reconstruction. *IEEE Transactions on Visualization and Computer Graphics* 5, 4, 349–359.
- BOISSONNAT, J., AND CAZALS, F. 2002. Smooth surface reconstruction via natural neighbour interpolation of distance functions. *Computational Geometry* 22, 1, 185–203.
- CALAKLI, F., AND TAUBIN, G. 2011. SSD: Smooth Signed Distance Surface Reconstruction. *Computer Graphics Forum 30*, 7. http://mesh.brown.edu/ssd.
- DEY, T. 2007. Curve and surface reconstruction: algorithms with mathematical analysis. Cambridge University Press.
- FLEISHMAN, S., COHEN-OR, D., AND SILVA, C. T. 2005. Robust moving least-squares fitting with sharp features. ACM Transactions on Graphics 24 (July), 544–552.
- HOPPE, H., DEROSE, T., DUCHAMP, T., MCDONALD, J., AND STUETZLE, W. 1992. Surface Reconstruction from Unorganized Points. In *Proceedings of ACM Siggraph*, Citeseer.
- KAZHDAN, M., BOLITHO, M., AND HOPPE, H. 2006. Poisson surface reconstruction. In *Proceedings of the fourth Eurographics symposium on Geometry processing*, Eurographics Association, 61–70.
- LORENSEN, W., AND CLINE, H. 1987. Marching Cubes: A High Resolution 3d Surface Construction Algorithm. 163–169.
- MANSON, J., PETROVA, G., AND SCHAEFER, S. 2008. Streaming surface reconstruction using wavelets. In *Computer Graphics Forum*, vol. 27, Wiley Online Library, 1411–1420.
- OHTAKE, Y., BELYAEV, A., ALEXA, M., TURK, G., AND SEI-DEL, H. 2005. Multi-level partition of unity implicits. In *ACM SIGGRAPH 2005 Courses*, ACM, 173.
- SCHAEFER, S., AND WARREN, J. 2005. Dual Marching Cubes: primal contouring of dual grids. In *Computer Graphics Forum*, vol. 24, Wiley Online Library, 195–201.