MOKSLAS – LIETUVOS ATEITIS
SCIENCE – FUTURE OF LITHUANIA

ISSN 2029-2341 print / ISSN 2029-2252 online
2011 3(1): 10–14 doi: 10.3846/mla.2011.002

Vaizdų technologijos T 111
Image Technologies T 111

Elektronika ir elektrotechnika
Electronics and Electrical Engineering

A SURVEY ON METHODS FOR RECONSTRUCTING SURFACES FROM UNORGANIZED POINT SETS

Vilius Matiukas

Vilnius Gediminas Technical University
E-mail: vilius.matiukas@ vgtu.lt

Abstract. This paper addresses the issue of reconstructing and visualizing surfaces from unorganized point sets. These can be acquired using different techniques, such as 3D-laser scanning, computerized tomography, magnetic resonance imaging and multi-camera imaging. The problem of reconstructing surfaces from their unorganized point sets is common for many diverse areas, including computer graphics, computer vision, computational geometry or reverse engineering. The paper presents three alternative methods that all use variations in complementary cones to triangulate and reconstruct the tested 3D surfaces. The article evaluates and contrasts three alternatives.

Keywords: unorganized point set, 3D surface reconstruction, Delaunay triangulation, complementary cones.

Introduction

To reconstruct a 3D object, its surface must be approximated using a set of 3D sample points (Remondino 2003; Pauly *et al*. 2004). Special devices for sampling surfaces such as 3D-laser scanners may be first used to generate the sets of the sample points required.

Data points produced by 3D-laser scanners are frequently unorganized. This implies that the ability to use them in 3D applications requires the computation of a polygon mesh (usually triangular-shaped) that best approximates the sampled surface. In turn, this calls for some connectivity structure to associate with the point set.

There are several algorithms for producing a 3D mesh from unorganized point sets. These algorithms differ in the methods employed as well as in their complexity and robustness. Unfortunately, many algorithms often fail to reproduce the connectivity (and sometimes even the topology) of the original object.

Hoppe *et al.* (1992) proposed an algorithm accepting an unorganized point set located on (or in the vicinity of) an unknown manifold to produce a simplified surface approximating the given manifold. The algorithm estimates a tangent plane at each sample using the *k*-nearest neighbors and uses the distance to the plane of the closest point as a signed distance function. Then, the marching-cubes algorithm is used to approximate the zero set of this function by piecewise-linear contours.

Amenta *et al.* (1998) introduced CRUST – another useful algorithm for reconstructing 3D objects – that was based on the Voronoi diagram. It was the first algorithm to theoretically ensure the topological correctness of the reconstructed surface. Eventually, CRUST became a theoretical foundation for the whole class of new algorithms (Amenta *et al.* 2000a, 2000b, 2001).

By combining the ideas present in the two algorithms above, another method for reconstructing surfaces based on coupling the Voronoi diagram with implicit functions was born (Boissonnat *et al.* 2000). The strength of this method lies in its ability to reconstruct surfaces having arbitrary topology and even using non-uniform sampling.

The Delaunay triangulation along with the Voronoi diagram are able to compensate for missing information on the geometry of an unorganized set of points provided that the sampling rate is adequate throughout the surface. One should be cautious, though, that undersampling (i. e. inappropriate choice of the sampling rate) tends to cause local changes in topology. In turn, those may result in holes appearing in reconstruction.

The Basic CoCone Method

The Delaunay triangulation is a good approximation of well sampled surface *S* from topological and geometrical points of view. However, such triangulation cannot be computed because the restricted Voronoi diagram cannot be computed without knowing S (Amenta *et al.* 2000a).

The Basic CoCone algorithm approximates the restricted Voronoi diagram and computes a set of candidate triangles from all Delaunay triangles. As an output, manifold surface approximating sampled surface S is extracted.

Each restricted Voronoi cell $V_p|S = V_p \cap S$ is almost flat if the input sample is sufficiently dense. The farthest Voronoi vertex p^+ in V_p is called the positive pole of sample p and $v_p = p^+ - p$ is pole vector for p. Pole vectors approximate the true normals of surface S at sample points P.

In this case, the CoCone of point p is set $C_p = \left\{ y \in V_p : \angle\left(\overrightarrow{py}, \mathbf{v_p}\right) \geq 3\pi/8 \right\}$, i.e. C_p is the complement of a double cone that is clipped within V_p. This double cone has p as the apex, pole vector $\mathbf{v_p}$ as the axis, an opening angle of $3\pi/8$ with the axis and y is the point in a Voronoi cell (Amenta *et al.* 2000a).

Fig. 1 shows Voronoi cell V_p and positive pole p^+. The double cone forming CoCone has the apex at point p and axis pp^+. Vector $\overrightarrow{pp^+}$ approximates surface's S normal at point p and Voronoi edge ab intersects CoCone. Its dual Delaunay triangulation is a candidate triangle.

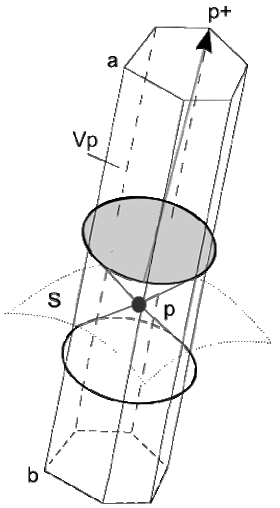

Fig. 1. Voronoi cell V_p and positive pole p^+
1 pav. Voronojaus ląstelė V_p ir teigiamas polius p^+

As an approximation to $V_p|S$, CoCones meet all Voronoi edges that are intersected by S. By computing all triangles dual to the Voronoi edges intersected by CoCones, all restricted Delaunay triangles are obtained. The set of these triangles is called candidate triangles T. They approximate the given unorganized point set closely to surface S.

Detection of Undersampling

The CoCone algorithm works nicely when input point set P densely samples surface S. In practice, the input often undersamples the surface. The CoCone algorithm computes many undesirable triangles near such regions.

Undersampling is defined as a set where points in P do not meet flatness condition stating that the point is sampled well if the following conditions are satisfied (Dey *et al.* 2001a):

– ratio: $r_p \leq \rho h_p$,

– normal: $\forall q$ with $p \in N_q$, $\angle\left(\mathbf{v}_p, \mathbf{v}_q\right) \leq \alpha$,

where $r_p = \max\left\{\left\|y - p\right\| : y \in C_p\right\}$ is the radius of CoCone C_p, $h_p = \left\|p^- - p\right\|$ is the height of Voronoi cell V_p and a set of $N_p = \left\{q \in P : C_p \cap V_p \neq\right\}$ is CoCone neighbors of p.

Ratio condition ensures that the Voronoi cell of point p is long and narrow along normal $\vec{n}_p$ of the point. Normal condition states that the direction of Voronoi cell extent is in phase with the direction of neighboring Voronoi cells extent.

Undersampling is detected in two steps. First, all points satisfying flatness condition are found. These points form set R, whereas others make set B. All points in R along with all their CoCone neighbors satisfy the ratio and normal conditions. On the second step, all points satisfying the ratio condition are located in B. If such a point has neighboring CoCone in R and their normal condition is met, then this point is included in R. The remaining points belong to the undersampled region (see Fig. 2).

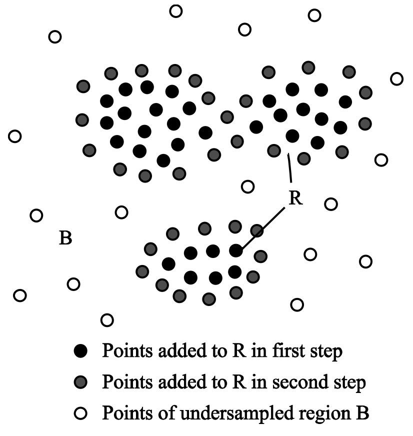
R
B
● Points added to R in first step
● Points added to R in second step
○ Points of undersampled region B

Fig. 2. Undersampling detection
2 pav. Nepakankamo diskretizavimo aptikimas

Finally, all triangles of the reconstructed surface, as an output of the CoCone algorithm, are verified. If any of these vertices of the triangles lies in the undersampled region, such triangle is removed from the reconstructed surface leaving holes on it.

The Tight Cocone Method

This complex method actually combines several algorithms. The basic CoCone method uses the Delaunay triangulation to approximate the surface by its poles. Then, the set of candidate triangles T is estimated by finding intersections between the CoCones and the edges of Voronoi cells. Next, all input points in P are divided into well-sampled and poorly-sampled regions applying the detection of undersampling. Triangles with vertices falling under poorly-sampled regions are then removed from T. Finally, the method detects and fills up all the holes (see the section titled The Umbrella Filter below) to produce a watertight surface. It should be noted that this process introduces no additional points, and the triangulated surface is generated by interpolating input sample points (Dey *et al.* 2003).

The Umbrella Filter

The structure of the vertices in well-approximated surfaces resembles that a topological disk is the so-called *umbrella*. All triangles in the umbrella have a common vertex at point p as along with two neighbors that share edges pw and pq (Fig. 3). The filter is used to iterate over candidate triangles and remove those not included in the umbrella.

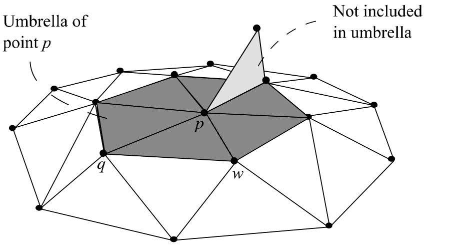

Fig. 3. Umbrella for point p
3 pav. Taško *p* skėtis

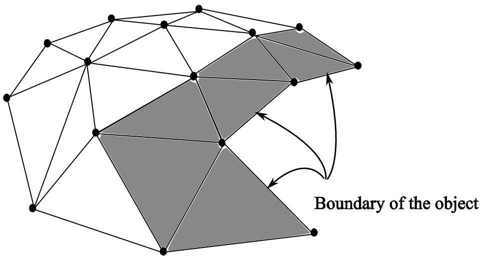

Fig. 4. Surface boundary detection
4 pav. Paviršiaus ribos nustatymas

A topological structure resembling an umbrella is also used for detecting the boundaries of the surface, i.e., holes in the undersampled regions. For a well-sampled surface, every point in 3D reconstruction has its own umbrella, as opposed to the case when holes show up (Fig. 4). The described method enables the detection of such cases and fills up the holes to make surface watertight (Amenta *et al.* 2001; Dey *et al.* 2003).

The Super CoCone Method

For unorganized point sets, reconstruction methods based on the Delaunay technique show good results both in theory and practice. However, such algorithms are too slow and cannot handle large data sets.

The Supper CoCone Algorithm (Dey *et al.* 2001b) extends the applicability of the CoCone algorithm to large data sets by using the divide-and-conquer approach. The basic idea behind the algorithm is reconstructing the 3D surface without computing the entire Voronoi diagram that consumes too much time and memory.

Instead, a local Voronoi diagram is computed for fewer sample points. Let $\widetilde{P} \subset P$ be such a subsample, where $p \in \widetilde{P}$. Such an approach requires satisfying only one condition, namely $\widetilde{P}$ must contain a set of all restricted Voronoi neighbors of p, R_p. Even though this ensures the right reconstruction of the surface, determining set R_p remains a problem without knowing surface S. As a remedy, R_p is overestimated by taking a subsample that has a cluster of sample points around p. For clustering, the octree subdivision method is used.

This method is applied in the following way. First, root node D in the octree containing all sample points is calculated. Second, the root node is divided recursively into eight equal subnodes. Each subnode needs to be divided further if it holds more sample points then the predefined threshold Δ. The procedure eventually concludes with the node of the lowest leaf holding a subset of sample points, which is smaller than the predefined value Δ.

Following the application of octree subdivision, the lowest adjacent leaf nodes are used for computing the Voronoi diagram. The rest of reconstruction is similar to the steps used in the CoCone algorithm described above.

Results

The algorithms were tested using a 2.8 GHz, 4GB RAM PC with Linux OS and 2.6.32 kernel preinstalled. For testing purposes, simulated and real data sets were used. Experimental data sets vary from simple geometrical

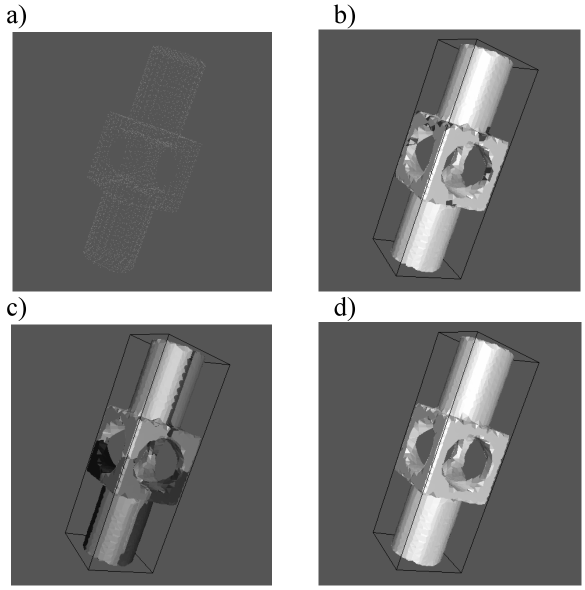

Fig. 5. The results of the point set reconstruction of the mechanical part using (a) CoCone (b), Super CoCone (c) and Tight CoCone (d) algorithms

5 pav. Mechaninės detalės nestruktūrizuoto taškų rinkinio (a) rekonstrukcijos rezultatai taikant dvikūgio (b), super dvikūgio (c) ir nepralaidaus dvikūgio (d) algoritmus

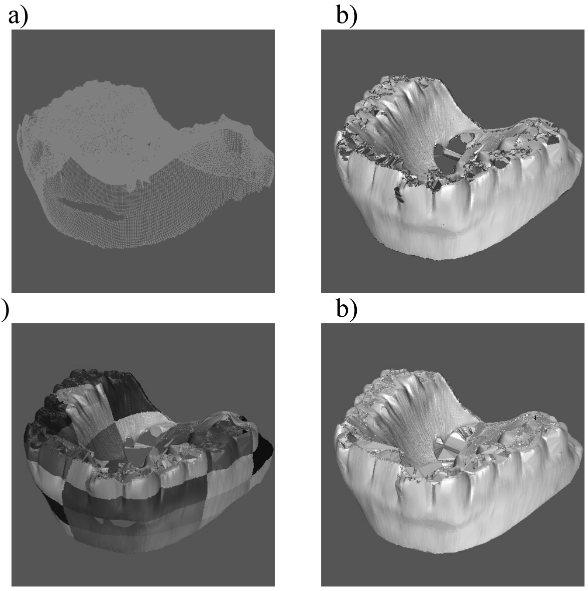

Fig. 6. The results of the point set reconstruction of the dental cast using (a) CoCone (b), Super CoCone (c), and tight CoCone (d) algorithms

6 pav. Dantų formos nestruktūrizuoto taškų rinkinio (a) rekonstrukcijos rezultatai taikant dvikūgio (b), super dvikūgio (c) ir nepralaidaus dvikūgio (d) algoritmus

shapes: from a mechanical part (Fig. 5 a) to a complex shape with high curvature: dental cast (Fig. 6 a).

Fig. 5 presents the reconstruction results of CoCone (Fig. 5 b), Super CoCone (Fig. 5 c) and Tight CoCone (Fig. 5 d) algorithms applied to the unorganized point set (Fig. 5 a) of the mechanical part. The point set of 4 100 points was simulated and a small-scale of noise of approximately 5 % was added.

The CoCone algorithm fails to reconstruct some sharp edges and in the final reconstruction leaves holes even on some planar regions of the surface (Fig. 5 b).

The Super CoCone algorithm generates results very similar to those of the CoCone algorithm. In this specific case, the Super CoCone algorithm uses the octree subdivision of 4-leaves nodes. The structure of subdivision is clearly seen in Fig. 5 (c) where the reconstruction of a different node is presented in a different color. The algorithm leaves holes in the regions where subdivision nodes intersect each other. This a cause of the points lack in the current set, thus the algorithm has no ability to rejoin reconstructed sub regions. An answer to this problem can be the increment of sample points or a deeper subdivision of the point set or both.

The tight CoCone algorithm was found to perform better with the current point set than the other algorithms and produced a water-tight surface.

Fig. 6 presents the results of dental cast reconstruction from the unorganized point set. This point set contains more than 300 K points. The shape of the sampled object is complex due to the curvature of the objects and none of the algorithms was able to gain satisfactory results. This problem is more of the point set collection hardware nature than reconstruction algorithms themselves.

Reconstruction times appear in Table 1. For a relatively small number of points (approximately 1 to 20 000) in an unorganized set, the results across all algorithms differ only in a few seconds. As the number of points in a set is increased, differences in the performance of algorithms become more evident. For example, during the reconstruction of the surface from 200 000 points, the Super CoCone algorithm was 6–7 times faster than the others. An interesting point is that this algorithm proved its ability to process very large data sets (on the order of several million) in a relatively moderate time. For instance, it only took more than 14 minutes to complete the reconstruction of a set containing one million points.

Table 1. Time for surface reconstruction

1 lentelė. Paviršiaus rekonstravimo laikas

Points	CoCone	Super CoCone	Tight Cocone
5 000	3.6 s	2.3 s	3.8 s
7 500	5.3 s	3.9 s	5.6 s
10 000	8.6 s	6.4 s	9.7 s
15 000	11.1 s	8.7 s	12.3 s
200 000	57.9 s	11.8 s	70.2 s
1 000 000	–	14 min 33 s	–

Conclusions

This paper described three alternative algorithms used for the reconstruction of 3D surfaces from unorganized point sets. These were titled as CoCone, Super CoCone and Tight CoCone. The evaluation of the algorithms revealed the following observations.

1. All alternatives discussed above theoretically ensured a topologically correct reconstruction of surfaces and provided relatively dense spacing between input point samples.
2. The CoCone algorithm failed to reconstruct sharp edges of the surface tested where a sharp edge was considered to be made of adjacent faces with an angle of $\pi/2$ or smaller between them.
3. The Super CoCone algorithm is the extension of the conventional CoCone algorithm suitable for large data sets. The Super CoCone algorithm proved to be quite efficient with large data sets. However, when tested on smaller data sets (on the order of a few thousand points), the algorithm could not reconstruct the surface at the intersections of subdivided regions.
4. The Tight CoCone algorithm managed to reconstruct a watertight surface, and this was not affected by the sampling rate. It should be noted, however, that the correctness of a geometrical approximation for the surface tends to deteriorate dramatically if sample points are spaced too far apart.
5. For sparse sets of unorganized points, reconstruction times for all algorithms considered differed by only in a few seconds. Differences became apparent for larger data sets. The Super CoCone algorithm was found to be 6–7 times faster than the other two. Also, it was only the Super CoCone algorithm that was able to reconstruct a large set involving one million points.

References

Amenta, N.; Bern, M.; Kamvysselis, M. 1998. A new Voronoi-based surface reconstruction algorithm, in *Proceedings of the 25th Annual Conference on Computer Graphics and interactive Techniques SIGGRAPH '98*. ACM, New York, NY, 415–421.

Amenta, N.; Choi, S.; Kolluri, R. K. 2001. The power crust, in *Proceedings of the Sixth ACM Symposium on Solid Modeling and Applications* (Ann Arbor, Michigan, United States). D. C. Anderson, Ed. ACM, New York, NY, 249–266. doi:10.1145/376957.376986

Amenta, N.; Choi, S.; Dey, T. K.; Leekha, N. 2000a. A simple algorithm for homeomorphic surface reconstruction, in *Proceedings of the Sixteenth Annual Symposium on Computational Geometry* (Clear Water Bay, Kowloon, Hong Kong, June 12–14, 2000). ACM, New York, NY, 213–222.

Amenta, N.; Choi, S.; Kolluri, K. R. 2000b. The Power Crust, Unions of Balls, and the Medial Axis Transform, in *Computational Geometry: Theory and Applications* (19): 127–153 [cited 26 May 2010]. Available from Internet <http://www.cs.utexas.edu/users/amenta/pubs/power.ps.gz>.

Boissonnat, J.; Cazals, F. 2000. Smooth surface reconstruction via natural neighbour interpolation of distance functions, in *Proceedings of the Sixteenth Annual Symposium on Computational Geometry* (Clear Water Bay, Kowloon, Hong Kong, June 12–14, 2000). ACM, New York, NY, 223–232.

Dey, T. K.; Goswami, S. 2003. Tight Cocone: A water-tight surface reconstructor, in *Proc. 8th ACM Sympos. Solid Modeling Applications,* 2003, 127–134.

Dey, T. K.; Giesen, J. 2001a. Detecting undersampling in surface reconstruction, in *Proceedings of the Seventeenth Annual Symposium on Computational Geometry* (Medford, Massachusetts, United States). D. L. Souvaine, Ed. ACM, New York, NY, 257-263. doi:10.1145/378583.378682

Dey, T. K.; Giesen, J.; Hudson, J. 2001b. Delaunay based shape reconstruction from large data, in *Proceedings of the IEEE 2001 Symposium on Parallel and Large-Data Visualization and Graphics* (San Diego, California, October 22–23, 2001). Parallel and large-data visualization and graphics. IEEE Press, Piscataway, NJ, 19–27. doi:10.1109/PVGS.2001.964399

Hoppe H., *et al*. 1992. Surface reconstruction from unorganized points, in *Proc. of Computer Graphics (SIGGRAPH '92),* Seattle, USA, 71–78.

Pauly, M.; Mitra, N. J.; Guibas, L. 2004. Uncertainty and Variability in Point Cloud Surface Data, in *Symposium on Point-Based Graphics,* The Eurographics Association, 77–84.

Remondino, F. 2003. From Point Cloud to Surface: The Modeling and Visualization Problem, in *Proc. of Int. Workshop on Visualization and Animation of Reality-based 3D Models,* Tarasp-Vulpera, Switzerland [cited 26 May 2010]. Available from Internet <http://www.photogrammetry.ethz.ch/general/persons/fabio/tarasp_modeling.pdf>.

PAVIRŠIŲ REKONSTRUKCIJOS IŠ NESTRUKTŪRIZUOTŲ TAŠKŲ RINKINIŲ METODŲ APŽVALGA

V. Matiukas

Santrauka

Šiame straipsnyje aprašomi ir lyginami trys metodai, skirti trimačių paviršių rekonstrukcijai iš nestruktūrizuotų jų taškų rinkinių. Paviršiaus aproksimacijai naudojamas tinklelis, sudarytas iš įvairių dvikūgių: paprasto, nepralaidžiojo ir superdvikūgio. Eksperimentiškai nustatyta, kad visi trys metodai tinka kokybiškai paviršiaus rekonstrukcijai, tik prieš tai reikia sudaryti pakankamai tankų pirminių taškų rinkinį. Be to, nustatyta, jog rekonstrukcijos nepralaidžiuoju dvikūgiu metodas tinka visiškai uždariems paviršiams, o rekonstrukcijos superdvikūgiu metodas savo sparta pranoksta kitus du metodus net 6–7 kartais. Šis metodas taip pat leidžia rekonstruoti paviršių ir esant itin dideliam (iki milijono) pirminių taškų rinkiniui.

Reikšminiai žodžiai: nestruktūrizuotas taškų rinkinys, trimačio paviršiaus rekonstrukcija, Delaunay trianguliacija, dvikūgis.