

Gan, Jacob Gah-tok

Spherical Maps:

Their Construction, Properties and Approximation

J.G. Gan
Nanyang Technological Institute
Singapore

T.C. Woo*, K. Tang
University of Michigan
1205 Beal Avenue
Ann Arbor, MI 48109-2117
U.S.A.

[1991?]

ABSTRACT

Gaussian map and its allied visibility map on a unit sphere find wide applications for orientating the workpiece for machining by numerical control machines and for probing by coordinate measurement machines. They also provide useful aids in computerized scene analysis, component design for manufacturing and the fabrication procedures. Spherical convex hulls and spherical circles are two geometric constructs used to approximate the Gaussian maps and the visibility maps. The duality and the complementarity of these spherical maps are examined so as to derive efficient algorithms.

Woo, A.C. (Anthony C.)
Tang, Kai

Geometry - - Data processing
Geometrical models.

* Please send correspondence to T.C. Woo at the above address.

BA P 1531

1 INTRODUCTION

The Gaussian map is a representation of the surface normals. Gauss introduced the concept of mapping the surface normals onto the surface of a unit sphere more than one-and-a-half centuries ago to define the local curvature of a given point [Hilbert 83]. The Gaussian map finds many applications. In numerical control machining applications, surface normals have long been used to provide tool orientations [Faux 79] such that offset surfaces may be generated [Chen 87, Farouki 86]. In component design and fabrication applications [Gan 89], the Gaussian map provides such informations as the directions which may potentially be used for deep drawing or stamping, and the lower bound on the number of inserts or cores necessary in the molding or casting process. In computer vision applications, the Gaussian map provides the basis for the generation of aspect graph [Sripradisvarakul 89] used in 3-D shape representation and for object recognition [Besl 88].

But it is not until recently that the tool geometry is incorporated in Gaussian maps. Surface normals can be "enhanced" by cutting tool geometries to yield the notion of a visibility map. Fast algorithms for intersection, union and convex hull on the sphere have been reported [Chen 90]. Opening a new vista in an allied application of coordinate measurement [Spyridi 90], a similar notion of an "accessibility map" is introduced based on measuring probe geometries. Uncovering the basic assumption that these derivatives of a Gaussian map imply a 3-axis machine application, [Tang 90] investigates the algorithmic aspect of a 4- and 5-axis applications in the context of spherical geometry. In the three aforementioned work, the existence of a visibility (or accessibility) map is assumed.

This paper reports the properties and the computational procedures for constructing the spherical maps. It generalizes the results for a specific surface representation [Kim 90]. In Section 2, the maps are defined and their construction and properties detailed. Because the Gaussian maps are not always convex, while the visibility maps are, the combination of two operators, (spherical) convex hull SCH and (spherical) dual, yields a pleasant surprise: taking the dual of the SCH of a Gaussian map gives the SCH of a visibility map. Since the visibility map is shown to be convex, the process, as reported in Section 3, gives a fast way to compute the visibility map by first approximating a Gaussian map. The paper goes one step further with another approximation - (spherical) circles. It shows in Section 4 that the in-circle of a visibility map can be computed from the circum-circle of a Gaussian map. There, two different operators, (spherical) Voronoi diagram and (spherical) complement are invoked.

2 SPHERICAL MAPS AND THEIR CONSTRUCTION

In this section, the Gaussian map and the visibility map are defined and their geometric constructions discussed.

2.1 Gaussian Map

Gauss introduced the notion of mapping of surface normals onto the surface of a unit sphere by means of "parallel normals" [Laugwitz 65, Hilbert 83, Calladine 86], where each point on the map is the result of the intersection of the surface normal vector, transferred parallelly such that it emanates from the center of the unit sphere, with the surface of the unit sphere. The region on the surface of the unit sphere so produced has come to be known as the *Gaussian map*. The following definition of the Gaussian map is from [do Carmo 76]:

Definition (Gaussian Map)

Let $S \subset \mathbb{R}^3$ be a surface with an orientation N . The map $N: S \rightarrow \mathbb{R}^3$ takes its values in the unit sphere $S^2 = \{(x,y,z) \in \mathbb{R}^3 \mid x^2+y^2+z^2=1\}$. The map $N: S \rightarrow S^2$, thus defined, is called the Gaussian map of S .

The geometric interpretation of a Gaussian map, or G-Map for abbreviation, is illustrated with the aid of Figure 2-1. The surface normals n_1, n_2 and n_3 of three sample points, p_1, p_2 and p_3 , on the surface S , are transferred to the unit sphere such that the directions do not change, and that the transferred vectors pass through the center of the sphere. The resulting points, p_1', p_2' and p_3' , on the G-map are obtained from the intersection of the transferred normal vectors with the surface of the unit sphere. When this is done for all the points on the surface of the unit sphere, the G-Map for the surface is produced.

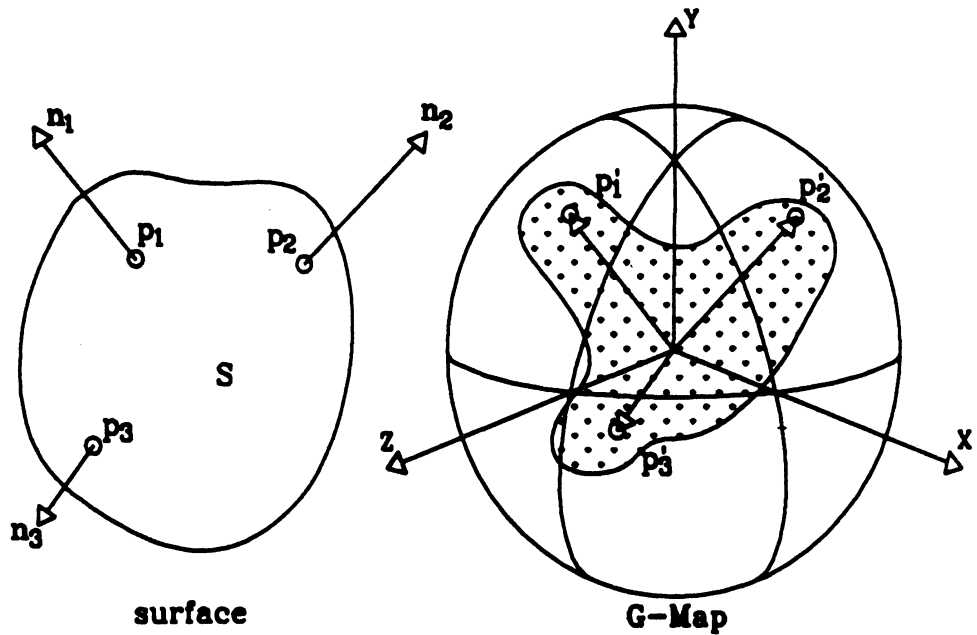


Figure 2-1 Definition of Gaussian Map

The G-Maps for some of the elementary surfaces are illustrated in Figure 2-2. For a plane, everywhere on the surface, the normal is the same. Therefore the G-Map for a plane is a single point. For a cylinder, the surface normals everywhere point in directions away from the axis of the cylinder. Therefore, the G-Map for a cylinder is a great circle. It follows that for a $1/2$ or a $1/4$ cylinder the corresponding G-Map should be $1/2$ or $1/4$ of a great circle. A sphere contains all the surface normals. Therefore its G-Map is also a sphere. Similarly, the G-Map of a hemisphere is another hemisphere. Due to the fact that the transfer of surface normal vectors is parallel, the position of the G-Map is the same as that of the progenitor hemispherical surface. For a cone, the surface normals everywhere incline at a fixed angle from the axis of the cone. Therefore, its G-Map is a small circle whose center corresponds to the axis of the cone, and whose radius (measured in term of the angle subtended by the circle at the center of the unit sphere) corresponds to the angle of inclination of the surface normal of the cone from its axis. Since the surface normals of a truncated cone are identical to those of a cone, the G-Map of a truncated cone is identical to that of a cone. Here, it is understood that the location of a G-Map on the unit sphere is a function of the orientation of a given surface in the Euclidean space. Procedures for constructing and approximating a G-Map are given in Appendix A.

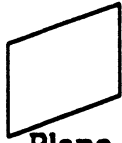
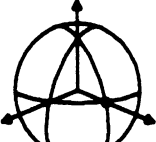

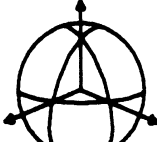
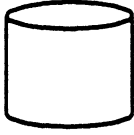
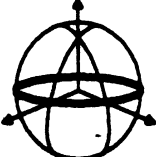


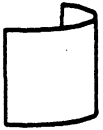
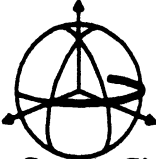



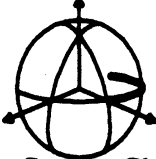


Surfaces	Gaussian Maps	Surfaces	Gaussian Maps
 Plane	 A Point	 Sphere	 Sphere
 Cylinder	 Great Circle	 Hemisphere	 Hemisphere
 1/2 Cylinder	 1/2 Great Circle	 Cone	 Small Circle
 1/4 Cylinder	 1/4 Great Circle	 Truncated Cone	 Small Circle

Figure 2-2 Surfaces and Their Gaussian Maps

2.2 Visibility Map

The normal of a point on a surface gives the direction in which the point is visible from infinity if it admits no intersection. But the same point may be visible from other directions as well. As illustrated in Figure 2-3, a tangent plane yields all other directions from which the point in question is locally visible. Based on this observation the notion of a visibility map is introduced as a representation of a collection of directions, each of which has the potential of viewing the complete surface.

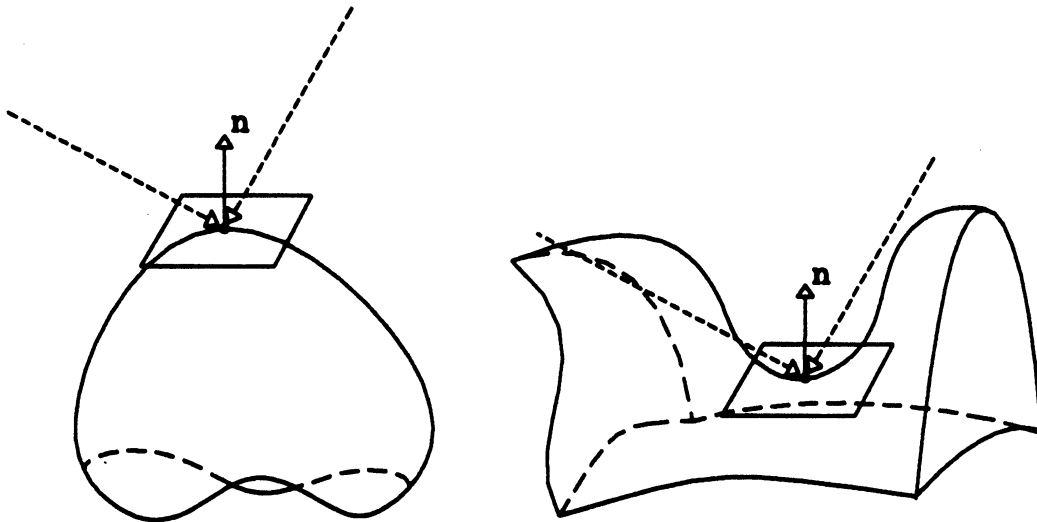


Figure 2-3 Tangent Plane and Visibility

Definition (Visibility Map)

A *visibility map* or *V-Map* of a surface S is the set of points p_i in S^2 , each of which differs from every normal n_j on S by at most 90 degrees. Symbolically,

$$\text{V-Map} = \{p_i \mid p_i \cdot n_j \geq 0; p_i \in S^2; n_j \in S^2\}.$$

An illustration of V-Maps with the aid of Figure 2-4 is in order. A plane has only a single normal, its G-Map is a single point in S^2 . The set of orientations which differs from this point by at most 90 degrees is a hemisphere having this point as the pole, or spherical center. The G-Map of a 1/2 cylinder is half-a-great-circle. Since the V-Map is the set of orientations which differ from this half-a-great-circle by at most 90 degrees, the V-Map is another half-a-great-circle, perpendicularly bisecting it. Since the G-Map of a sphere is a sphere, the G-Map of a hemisphere is also itself. The V-Map of a hemisphere is a point at its pole - it is the only point from which the normals of the hemisphere differ by no more than 90 degrees.

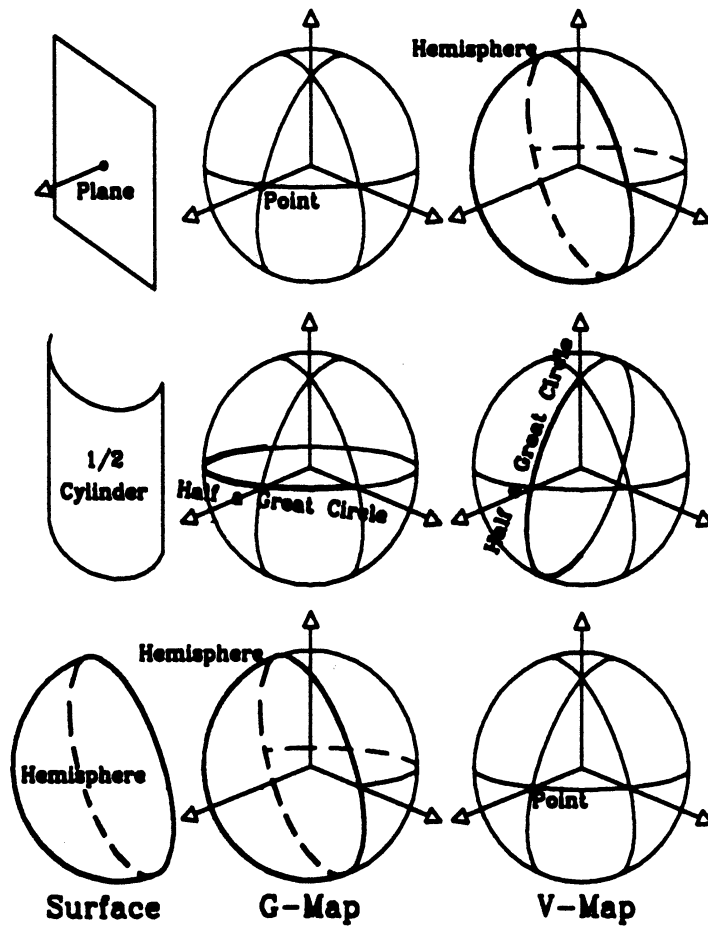


Figure 2-4 Visibility Maps for Plane, Cylinder and Hemisphere

The V-Map is a set of points in S^2 such that each point in the set differs from every point in the G-Map by at most 90 degrees. The geometric construction of a V-Map given a G-map (for an arbitrary parametric surface) is now shown.

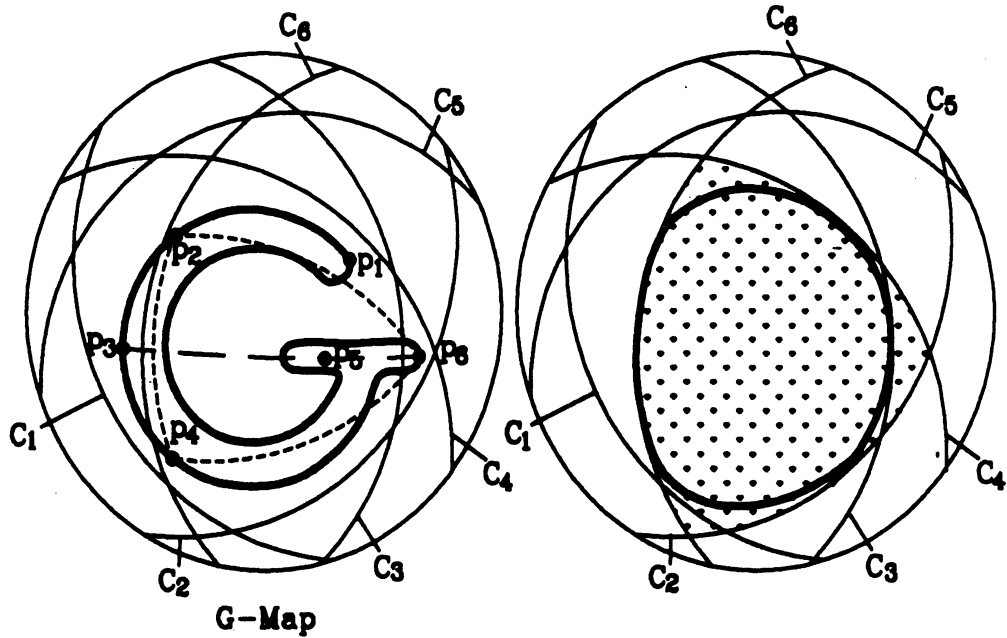


Figure 2-5 Construction of Visibility Map

The V-Map of a surface can be constructed by computing the intersection of hemispheres, each having as its pole a point on the G-Map, i.e.

$$V\text{-Map}(S) = \bigcap_{p_i \in G\text{-Map}(S)} \text{hemi}(p_i)$$

Suppose the figure "G" in Figure 2-5 represents the G-Map of a given surface. Take a point p_1 on the G-map. The set of points in S^2 that deviate from p_1 by at most 90 degrees is a hemisphere $\text{hemi}(p_1)$ bounded by the great circle C_1 having the point p_1 as its pole. Similarly, corresponding to the points p_2 to p_6 , the sets of points that deviate from each of them by at most 90 degrees is each a hemisphere, denoted by $\text{hemi}(p_2)$ to $\text{hemi}(p_6)$ bounded by the great circles C_2 to C_6 respectively. The intersection of the hemispheres corresponding to the three points p_2 , p_4 and p_6 is shown by the dotted figure on the right. The intersection of all the hemispheres $\text{hemi}(p_i)$ for all points p_i in "G" yields the oval shape indicated in bold.

The procedure for constructing a V-Map is immediate.

Procedure V-Map(G-Map)

- step 1 $V\text{-Map} := \text{complete sphere}$
- step 2 For each sample point p_i on the G-Map Do
 $V\text{-Map} := V\text{-Map} \cap \text{hemi}(p_i)$

While the complexity of step 2 of *Procedure V-Map* appears to be $O(n^2)$, where n is the number of hemispheres or points approximating the G-Map, it is $O(n \log n)$ by rewriting it in a divide-and-conquer form as *Procedure V-Map**. It and a recursive *Function Partition* takes as input a set of points P in S^2 and returns as output the intersection of hemispheres whose spherical centers are the input points, are given below.

Procedure V-Map (G-Map)*

- step 1 $V\text{-Map} = \text{Partition}(G\text{-Map})$

Function Partition(P)

- step 1 If $\text{cardinality}(P) > 1$ Then
- step 2 Divide P into P_1 and P_2 of approximately equal sizes
- step 3 $\text{Part}_1 := \text{Partition}(P_1)$
- step 4 $\text{Part}_2 := \text{Partition}(P_2)$
- step 5 Return $\text{Part}_1 \cap \text{Part}_2$
- step 6 Else Return $\text{hemi}(P)$

Though *Procedure V-Map** appears straightforward, the number of sample points it takes is from the entire G-Map. In the following Section, it is shown that only the points from the boundary of a G-Map - and, in particular, only the points on the convex hull of a G-Map - are necessary and sufficient.

3 EFFICIENT COMPUTATION OF V-MAP

The strategy for computing a V-Map efficiently from a G-Map is, as illustrated in Figure 3-1, by taking the spherical convex hull SCH of a G-Map and then by taking its dual: for each vertex in the SCH(G-Map) there corresponds an edge in the spherical convex hull of the V-Map, SCH(V-Map).

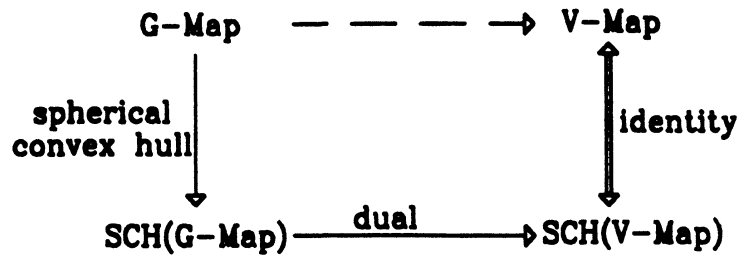


Figure 3-1 Finding Visibility Map

This strategy yields the SCH(V-Map). As the rest of this Section is devoted to finding the spherical convex hull and to finding the dual, the convexity of the V-Map is firstly established, i.e. SCH(V-Map) is identical to V-Map.

Definition (Spherical Convexity)

A set of points in S^2 is (*spherically*) *convex*, if for any two points p_1 and p_2 in the set, the segment p_1p_2 (the shorter of the two great arcs joining p_1 and p_2) lies entirely in it.

From the construction:

$$V\text{-Map}(S) = \bigcap_{p_i \in G\text{-Map}(S)} \text{hemi}(p_i)$$

a V-Map is the intersection of hemispheres, each corresponding to a point in the given G-Map. A hemisphere is, by definition, convex. (Its correspondence to a half plane, by central projection, is alternatively given in Appendix B.) As the intersection of convex sets is always convex, a V-Map is therefore convex. Hence, it is identical to its convex hull. The discussion now moves to the first of two operations: finding the spherical convex hull of a G-Map.

3.1 Spherical Convex Hull

It may be interesting to note that the points in the interior of a G-Map do not contribute to the construction of a V-Map. In Figure 2-5, the point p_5 is in the interior of a G-Map. Its corresponding hemisphere $hemi(p_5)$ bounded by great circle C_5 is a superset of the eventual V-Map, no part of the boundary of which comes from C_5 . By contrast, p_3 and p_6 , which are on the boundary of the G-Map in Figure 2-5, give rise to the V-Map. These observations are not difficult to rationalize. Suppose p_3 and p_6 are the end points of a segment (of a great circle) and p_5 lies somewhere in the segment. As long as the segment is less than or equal to a semi great circle, the intersection of the two hemispheres bounded by C_3 and C_6 will be completely contained in C_5 . An analog in the plane is shown in Figure 3-2.

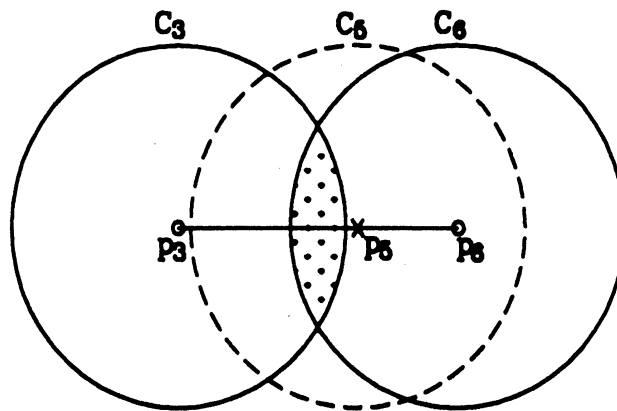


Figure 3-2 Contributions to V-Map

That only the points on the boundary of a G-Map contributes to a V-Map is formalized by the following.

Lemma 3-1

For the determination of the V-Map, the set of points on the boundary of the G-Map is sufficient.

Proof:

Let p_i be any point in the interior of the G-Map and C any great circle through it. There always exist on C two other boundary points p_b and p_b' on either side of p_i , as shown in Figure 3-3, such that

$$\text{hemi}(p_b) \cap \text{hemi}(p_b') \subseteq \text{hemi}(p_i)$$

Since a V-Map comes from the intersection of hemispheres and since $\text{hemi}(p_b) \cap \text{hemi}(p_b') \cap \text{hemi}(p_i) = \text{hemi}(p_b) \cap \text{hemi}(p_b')$, only boundary points p_b and p_b' contribute to the V-Map.

Q.E.D.

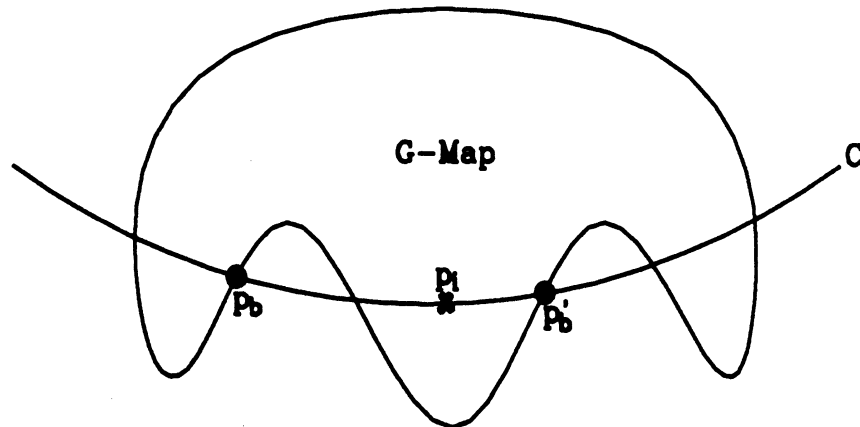


Figure 3-3 Contributions from Boundary Points of Gaussian Map

For the construction of a V-Map, Lemma 3-1 reduces the range of data drastically from the domain of an entire G-Map to that of its boundary. A further reduction is made possible by invoking the notion of convexity. It is useful to note that the points on the boundary of a G-Map lying inside its convex hull do not contribute to the construction of a V-Map.

Definition (Spherical Convex Hull)

The *spherical convex hull SCH* of a set of points in S^2 is the boundary of the smallest convex set containing it.

A reasoning similar to the proof for Lemma 3-1 proceeds as follows, with the aid of Figure 3-4 in which the portion of the boundary that coincides with the convex hull is darkened.

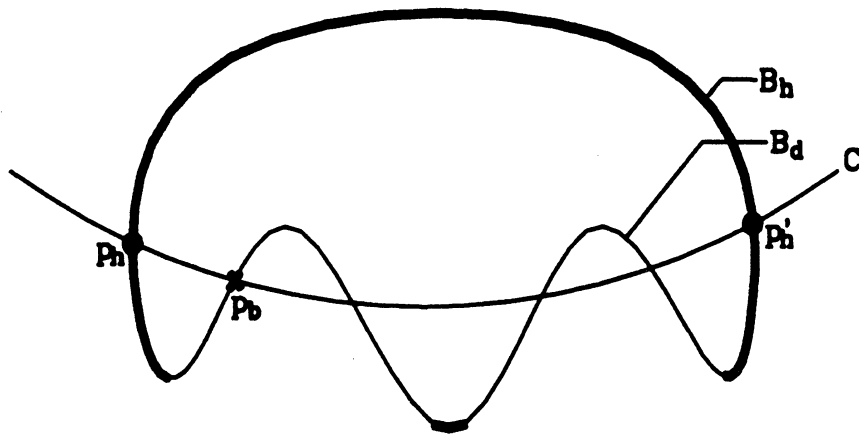


Figure 3-4 Contributions from Convex Hull Points

Let the boundary B of a G-Map be divided into two disjoint subsets: the portion that lies on the convex hull B_h and the portion that lies on the convex deficiency B_d . Let p_b be any point in B_d . There always exist a great circle C through p_b such that C intersects B_h at exactly two points: p_h and p_h' on either side of p_b . The point in the center p_b does not contribute to the V-Map but the other two on B_h do. The following lemma ensures further reduction of the range of data.

Lemma 3-2

For the determination of the V-Map, the set of points on the boundary of a G-Map that coincides with its convex hull is necessary.

The Procedure for determining the spherical convex hull of a G-Map is given in Appendix B.

3.2 Duality

While the notion of convex hull helps reduce the range of data (by providing a small n), the notion of duality enables efficiency in finding the V-Map (from a G-Map by providing an $O(n)$ algorithm).

Definition: (Duality of Spherical Convex Hulls)

Two spherical convex hulls, SCH_1 and SCH_2 , are *dual* to each other if a vertex V_i of SCH_1 corresponds to an edge E_i in SCH_2 such that E_i is a great arc having V_i as the pole, and the vertices in SCH_1 are in the same order as the corresponding edges in SCH_2 ; and vice versa.

The two spherical convex hulls given in Figure 3-5 are dual to each other. Great arcs C_1 to C_4 have their poles at vertices p_1 to p_4 respectively, and they are ordered in the same sense.

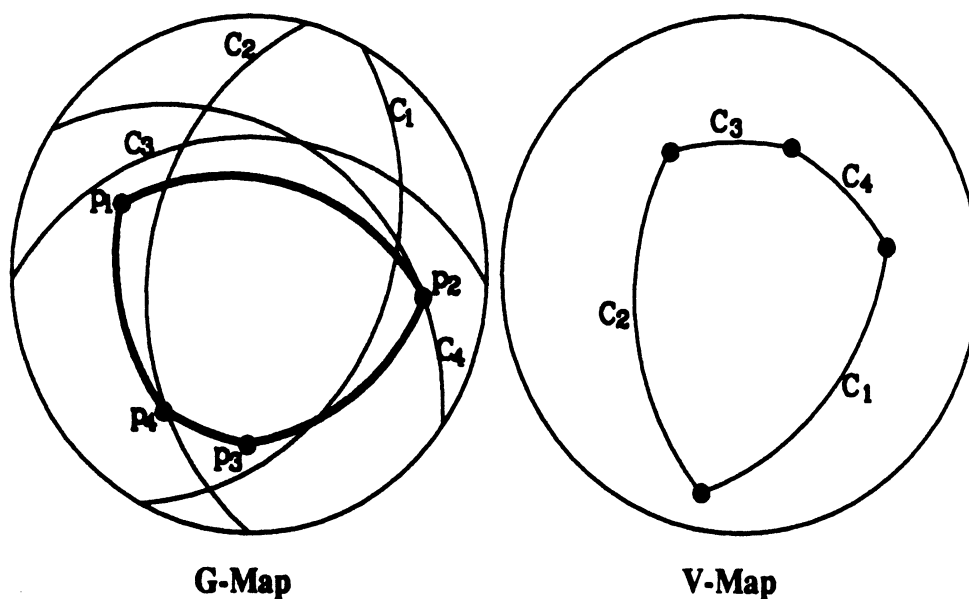


Figure 3-5 Duality of Spherical Convex Hulls

Lemma 3-3

SCH(V-Map) is the dual of SCH(G-Map)

Proof:

Take a spherical convex hull whose vertices are p_1, p_2, p_3 and p_4 , in clockwise order (see Figure 3-5). The set of points in S^2 which deviate from vertex p_1 by at most 90 degrees is a hemisphere centered on p_1 (shown as hemisphere bounded by the great circle C_1). This establishes the duality between a vertex in SCH(G-Map) and an edge in SCH(V-Map). The ordering of the edges in SCH(V-Map) could be different from that of the vertices in SCH(G-Map) if some of the vertices p_i were not used. By Lemmas 3-1 and 3-2, the vertices of SCH(G-Map) are necessary and sufficient. In other words, if the duals of the vertices of SCH(G-Map) are taken in turn, the intersection of the great circles must yield edges for SCH(G-Map) in the order taken.

Now, SCH(V-Map) is the V-Map itself because hemispheres are convex sets and the intersection of convex sets is convex.

The SCH(V-Map) is therefore the dual of the SCH(G-Map).

Q.E.D.

Lemma 3-3 suggests that the V-Map of a set of points in S^2 can be obtained by first finding the spherical convex hull of the G-Map and then dualizing the vertices of its spherical convex hull. It becomes clear then, given v convex hull vertices of a G-Map, a V-Map can be constructed in $O(v)$ time as the corresponding hemispheres can be intersected sequentially.

Procedure V-Map(G-Map)

step 1 Determine SCH(G-Map)

step 2 Determine the dual of SCH(G-Map) to get SCH(V-Map)

The overall complexity of the *Procedure V-Map* is however $O(n \log n)$ because, while *step 2* is $O(v)$ where v is the number of vertices in the convex hull of G-Map *step 1* is $O(n \log n)$ which dominates the time complexity.

4 Spherical Circles as Approximations

Though a V-Map is convex, an in-circle may be used as a further approximation.

Definition (In-Circle)

The largest circle within the boundary of a spherical region in S^2 .

The in-circle is appropriate as an approximation to the boundary of a V-Map because it is the largest circle within the boundary of the V-Map and it does not enclose any point not lying within the V-Map. Finding an in-circle of an n -sided spherical polygon efficiently, however, is difficult as there can be combinatorially many solutions: up to $\binom{n}{3}$ circles may be tangential to three of these n -edges and not all of them are valid candidates for the in-circle.

A novel approach to the determination of the In-Circle(V-Map) is introduced. It is computed indirectly, in much the same way as the V-Map is computed from the dual of the convex hull of a G-Map, by way of the complement of the circum-circle for a G-Map.

Definition (Circum-Circle)

The *Circum-Circle* of a set of points P in S^2 is the smallest circle in S^2 bounding P .

The strategy for approximating a V-Map is illustrated in Figure 4-1.

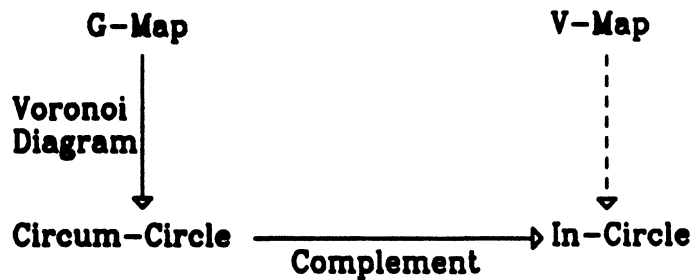


Figure 4-1 Finding the In-Circle of a Visibility Map

4.1 Finding the Circum-Circle

An immediate method for finding the circum-circle for a set of points is to determine the diameter of the set.

Definition (Diameter of a Set of Points)

The *diameter* of a set of points P in S^2 is the maximum distance between two of them, measured along a great arc.

Figure 4-2 (a) illustrates the diameter of a set of points. Since the SCH of a set of points in S^2 can be computed in time $O(n \log n)$ where n is the number of points in the set, and given the SCH the diameter of the set can be computed in time $O(v)$ [Preparata 85] where v is the number of vertices in SCH, the diameter of a set of points can be computed in time $O(n \log n)$. As noted from Figure 4-2 (b), however a circle of diameter d may not cover the entire set.

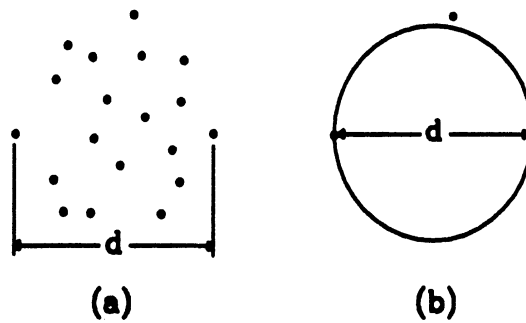


Figure 4-2 Diameter and Point Inclusion

That it requires additional testing for point inclusion in $O(n)$ time notwithstanding, merely finding the diameter for a circum-circle is not sufficient. A remedy is sought.

The concept of a *Farthest Point Voronoi diagram* is invoked to develop the procedure for determining the three-point circum-circle of a set of points in S^2 . A more familiar version of the closest point Voronoi diagram is given first.

Definition (Spherical Closest Point Voronoi Diagram)

The *spherical closest point Voronoi diagram* of a finite set points P in S^2 is a partitioning of S^2 so that each region of the partition is the locus of points which are closer to one member of P than to any other member.

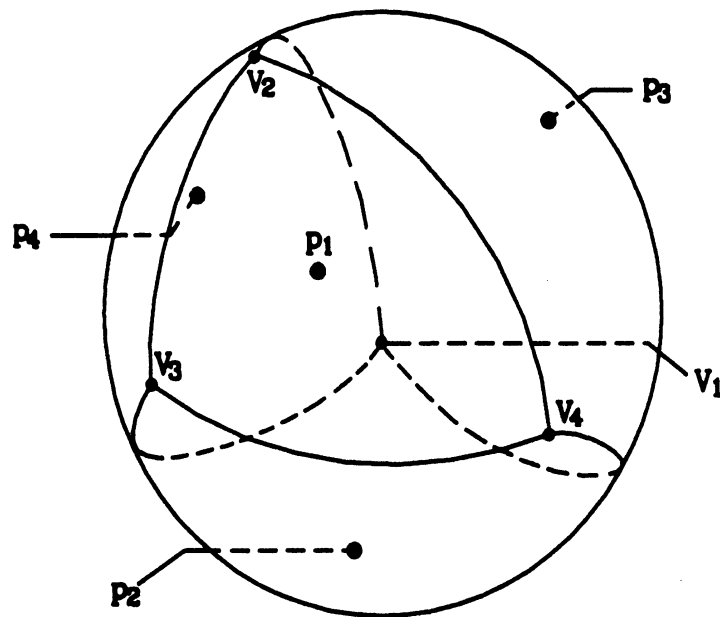


Figure 4-3 Spherical Closest Point Voronoi Diagram

A closest point Voronoi diagram for a set of four points is illustrated in Figure 4-3. Every point within the spherical polygonal region whose vertices are V_2 , V_3 and V_4 is closer to the point p_1 than to any other point in the set. Similarly, point p_2 is closest to polygonal region V_1 - V_4 - V_3 , point p_3 closest to polygonal region V_1 - V_2 - V_4 , and point p_4 closest to polygonal region V_2 - V_1 - V_3 .

The complement to the spherical closest point Voronoi diagram is defined below:

Definition (Farthest Point Voronoi Diagram)

The *(spherical) farthest point Voronoi diagram* of a finite set of points P in S^2 is a partitioning of S^2 so that each region of the partition is the locus of points which are farther from one member of P than from any other member.

A farthest point Voronoi diagram (FPVD) for the same set of four points as that for the closest point Voronoi diagram is illustrated in Figure 4-4. Every point within the spherical polygonal region whose vertices are V_2 , V_3 and V_4 is farthest from the point p_1 than from any other point in the set. Similarly, every point in the polygonal region V_1 - V_3 - V_4 is farthest from point p_2 , every point in the polygonal region V_1 - V_4 - V_2 farthest from point p_3 , and every point in the polygonal region V_1 - V_2 - V_3 farthest from point p_4 .

The $O(n \log n)$ algorithm [Shamos 75] for determining the closest point Voronoi diagram can be used to determine the FPVD. This is based on the observation that the FPVD for a set of points P is actually the closest point Voronoi diagram of a set of antipodal points, each of which lies diametrically on the opposite side of the sphere to a point in the set P [Anderson 85].

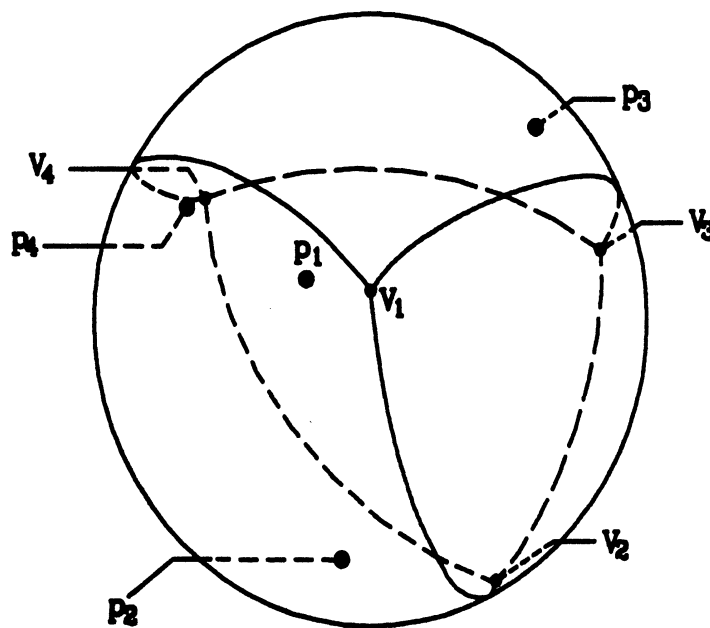


Figure 4-4 Spherical Farthest Point Voronoi Diagram

An FPVD vertex V_i has the property that it is equally distant to the set of three points p_i that contribute to the construction of V_i . Thus there exists a circle centered at a FPVD vertex V_i which passes through p_i .

Lemma 4-1

The circle centered on a vertex of a FPVD, and has a radius equal to the distance from this vertex to any of the three points that define it, encloses all the points in the set from which the Voronoi diagram is constructed, i.e. all the points lie on the same side of the circle as the vertex.

Proof:

Each polygon in the FPVD is the locus of a point that is farther from a given point in the set than from any other point in the set. Therefore, an FPVD vertex, the intersection of three adjacent polygons, is farther from each of the three points in the set than from all the other points in the set. That means, there is no point in the set outside the circle defined by the three points that determine the FPVD vertex.

Q.E.D.

It is important to note that the number of such three-point circles does not reach the naive bound of nC_3 . By the definition of spherical Voronoi diagram on a set of n points in S^2 , S^2 is partitioned into n regions (spherical polygons). The dual of a Voronoi diagram is the triangulation of S^2 [Preparata 85]. Therefore, the triangulated S^2 has n vertices. By the application of the celebrated Euler's formula [Bollobas 79, Swamy 81] to triangulation, triangulating a set of n vertices in S^2 partitions S^2 into $2n-4$ spherical triangles. $2n-4$ triangulated regions dual to $2n-4$ vertices in the FPVD Hence, there are $2n-4$ such three-point circles.

The determination of the circum-circle of a set of points P in S^2 can be summarized by the following algorithm:

Procedure Circum-Circle(G-Map)

step 1 Find diameter d of the set of sample points P in the G-Map

step 2 If spherical circle with diameter d covers P

then Circum-Circle is found

otherwise, go to step 3

step 3 Find "Spherical Farthest Point Voronoi Diagram" of P

step 4 Circum-Circle = Smallest(three-point circles)

step 1 takes $O(n \log n)$ time, step 2 $O(n)$ time, step 3 $O(n \log n)$ time, and step 4 $O(n \log n)$ time. Therefore, the overall complexity is $O(n \log n)$.

4.2 Complementarity

Definition (Complementarity of Spherical Circles)

Two spherical circles, C_C and C_I , are complementary to each other if they have the same spherical centers and the angle subtended by C_C at the center of the unit sphere and that subtended by C_I are 180 degrees complement of each other.

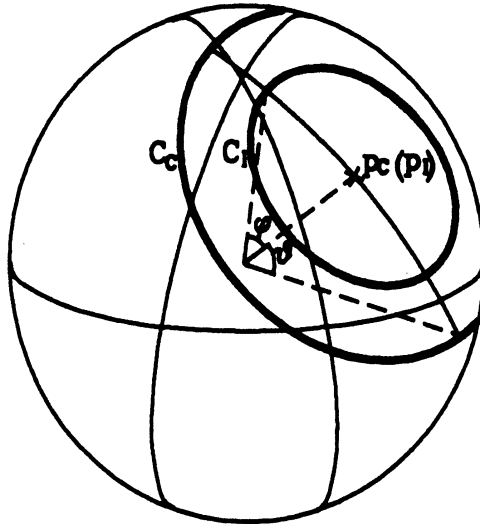


Figure 4-5 Complement Circles

Let C_C be the circum-circle approximating the G-Map and C_I the corresponding in-circle approximating the V-Map. The following results are immediate by definition of the V-Map.

Lemma 4-2

If the centers of C_C and C_I are p_C and p_I , and the angles subtended by C_C and C_I are 2ϕ and 2θ respectively. Then

- (i) $p_C = p_I$; and
- (ii) $\phi + \theta = 90 \text{ deg.}$

The in-circle C_I approximating the V-Map is computed by taking the complement of the circum-circle C_C for the corresponding G-Map. Having asserted the existence of two complementary circles, it is necessary to show that C_I thus obtained is indeed tangential to the V-Map, with the aid of Figure 4-6.

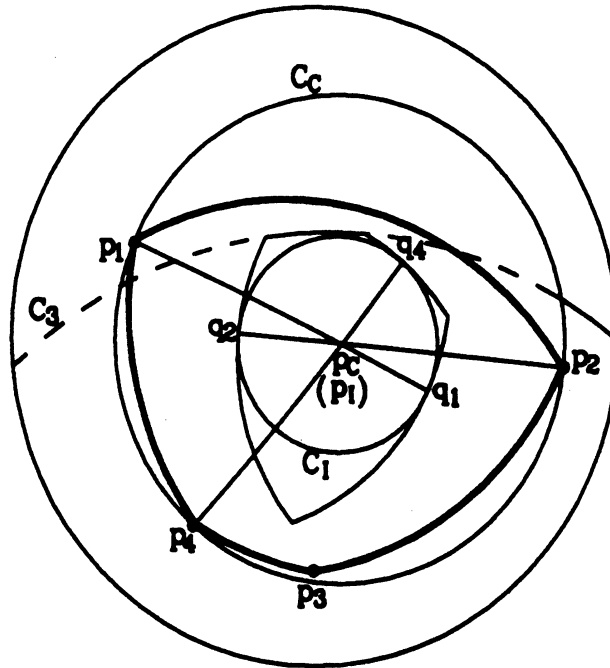


Figure 4-6 Tangency of C_I and Complementarity to C_C

Lemma 4-3

Let p_i be a vertex on the G-Map which also lies on C_C . Then, there exists a point q_i such that it is the point of tangency between C_I and the great arc centered at p_i .

Proof:

By Lemma 4-2, since p_i is a point on C_C , there exists a point q_i on C_I such that the great arc $p_i q_i$ subtends an angle of 90 degrees at the center of the unit sphere, and the great arc $p_i q_i$ passes through p_C (and p_I). The point q_i therefore lies on the great arc centered at p_i . The result follows.

Q.E.D.

While Lemma 4-3 shows the tangency of C_I to a V-Map, it is insufficient unless C_I is also shown to be an in-circle to the V-Map, that is, C_I is a conservative approximation to a V-Map.

Lemma 4-4

None of the great arcs, centered at vertices p_j not lying on C_C , intersects C_I .

Proof:

Let p_j be a vertex on the G-Map which does not also lie on C_C . Since p_j lies within C_C , everywhere on the circle C_I differs from p_j by less than 90 degrees. Therefore, the edge on the V-Map which forms part of the great arc centered at p_j does not intersect C_I .

Q.E.D.

In Figure 4-6, p_3 does not lie on C_C . Its corresponding great circle C_3 , while giving rise to the V-Map, does not intersect the in-circle C_I approximating the V-Map.

That the in-circle to a V-Map can be obtained by taking the complement (via Lemma 4-2) of the circum-circle of a G-Map (by Procedure Circum-Circle) is now ready.

Lemma 4-5

In-Circle(V-Map) is the complement of circum-circle(G-Map).

Proof:

By Lemma 4-3, C_I is the circle which is tangent to edges of the V-Map which are parts of the great arcs, centered at the vertices of the G-Map lying on the Circum-Circle(G-Map). By Lemma 4-4, C_I lies fully within the SCH(V-Map). This completes the proof.

Q.E.D.

Lemma 4-5 above gives the basis that the in-circle of the V-Map can be found by complementing the circum-circle of the G-Map, as an alternative to finding the spherical convex hulls of the G-Map and the V-map.

Procedure In-Circle(G-Map)

step 1 Determine Circum-Circle of G-Map

step 2 In-Circle(V-Map) is given by the Complement of the Circum-Circle found in step 1

4 SUMMARY

Procedures for computing a G-Map from a given surface and for computing a V-Map from a G-Map have been presented. In addition to the procedure V-Map*, which takes the set intersection of hemispheres, each of which corresponding to a point in the G-Map, two efficient schemes for computing a V-Map by approximation are given.

Since a V-Map is convex, it is identical to its spherical convex hull SCH(V-Map). It is shown in Lemma 3-3 that SCH(V-Map) can be obtained by taking the dual of the SCH(G-Map): for each point p_i on the convex hull of a G-Map there corresponds an edge C_i in the convex hull of a V-Map, and such C_i is a part of the great circle centered at p_i on the Gaussian sphere.

An in-circle is proposed as a further approximation to a V-Map. Lemma 4-5 eases the difficult computation by complementing the circum-circle of a G-Map: that the two circles have identical centers on the sphere with solid angles 180 degrees complement to each other. A three-point circum-circle for a G-Map is obtained from its farthest point Voronoi diagram.

APPENDIX A

Construction and Approximation of G-Map

A surface S may be represented by parametric equations for its three orthogonal components [Faux 79, Foley 82]:

$$S(u,v) = X(u,v) i + Y(u,v) j + Z(u,v) k$$

where u and v are the parameters, and i , j and k the unit vectors in the x , y and z axis respectively.

Let S_u and S_v be the first derivatives of S with respect to the two parameters u and v respectively. Then, the unit surface normal, \mathcal{N} , is given by the normalized cross-product of S_u and S_v , as follows:

$$\mathcal{N} = \frac{S_u \times S_v}{|S_u \times S_v|}$$

If S is of the order m in parameter u and n in parameter v , then the numerator of \mathcal{N} is of the order $2m-1$ in parameter u and $2n-1$ in parameter v , and the denominator of \mathcal{N} is the square root of an expression which is of the order $2(2m-1)$ in u and $2(2n-1)$ in v . Therefore, the analytical expression of \mathcal{N} is of a much higher order and complexity than that of the progenitor surface.

For example, for a bicubic parametric surface $S(u,v)$ whose X -component is:

$$\begin{aligned} X(u,v) = & p_{00x} & + & p_{01x}v & + & p_{02x}v^2 & + & p_{03x}v^3 & + \\ & p_{10x}u & + & p_{11x}uv & + & p_{12x}u^2v & + & p_{13x}u^3v & + \\ & p_{20x}u^2 & + & p_{21x}u^2v & + & p_{22x}u^2v^2 & + & p_{23x}u^2v^3 & + \\ & p_{30x}u^3 & + & p_{31x}u^3v & + & p_{32x}u^3v^2 & + & p_{33x}u^3v^3 \end{aligned}$$

where u and v are the parameters which vary from 0 to 1, the corresponding derivatives and cross-products are :

$$\begin{aligned} X_u(u,v) = & p_{10x} & + & p_{11x}v & + & p_{12x}v^2 & + & p_{13x}v^3 & + \\ & 2p_{20x}u & + & 2p_{21x}uv & + & 2p_{22x}u^2v & + & 2p_{23x}u^3v & + \\ & 3p_{30x}u^2 & + & 3p_{31x}u^2v & + & 3p_{32x}u^2v^2 & + & 3p_{33x}u^2v^3 \end{aligned}$$

$$\begin{aligned}
X_v(u,v) = & p_{01x} + 2p_{02x}v + 3p_{03x}v^2 + \\
& p_{11x}u + 2p_{12x}uv + 3p_{13x}uv^2 + \\
& p_{21x}u^2 + 2p_{22x}u^2v + 3p_{23x}u^2v^2 + \\
& p_{31x}u^3 + 2p_{32x}u^3v + 3p_{33x}u^3v^2
\end{aligned}$$

$$\begin{aligned}
S_u \times S_v(u,v) = & (Y_u(u,v)Z_v(u,v) - Z_u(u,v)Y_v(u,v)) \mathbf{i} + (Z_u(u,v)X_v(u,v) - X_u(u,v)Z_v(u,v)) \mathbf{j} \\
& + (X_u(u,v)Y_v(u,v) - Y_u(u,v)X_v(u,v)) \mathbf{k}
\end{aligned}$$

While \aleph is of high order in u and v , the G-Map, taking values on the unit sphere, is quadratic in some other parameters. However, reparameterizing \aleph to give an analytical expression to the G-Map is non-trivial. Rather than representing a G-Map by its analytical form, it may be approximated by a set of sample surface normals. Sampling may be performed by first approximating the G-Map by a series of constant parametric curves, each of which, in turn, is approximated by a series of sample points. If the sample points are chosen at uniform parametric intervals, the forward-differencing method [Stanton 60, Gerald 84] may be employed to determine the polynomial $(S_u \times S_v)$ at each of the consecutive sample points, which when normalized gives the value of the surface normal. The tolerance (*tol*) on the maximum "gap" in the values of sample normals has to be satisfied for the approximation of the G-Map to be acceptable. This can be achieved by ensuring that the interval (δu or δv) of sampling in the parametric space is such that its product with maximum rate of change of the surface normal with respect to the parameter u or parameter v (\aleph_{uMax} or \aleph_{vMax}) is not greater than *tol*, i.e.

$$\begin{aligned}
\delta u \times \aleph_{uMax} & \leq \text{tol}, \text{ and} \\
\delta v \times \aleph_{vMax} & \leq \text{tol}
\end{aligned}$$

A procedure that generates sample normals from a given surface S , satisfying the tolerance on the maximum gap in the values of the sample normals, approximating the G-Map is summarized below:

Procedure G-Map(S)

- step 1 Determine the coefficients of $(S_u \times S_v)$*
- step 2 Compute \aleph_{uMax} and \aleph_{vMax}*
- step 3 $\delta u = \text{tol} / \aleph_{uMax}$ and $\delta v = \text{tol} / \aleph_{vMax}$*
- step 4 Approximate $(S_u \times S_v)$ by a series of constant- u curves spaced δu apart*
- step 5 For each of the constant- u curves Do*
 - step 5.1 Compute initial forward differences based on δv increment*
 - step 5.2 For each increment of v by δv Do*
 - step 5.2.1 Determine new value of $(S_u \times S_v)$ by adding the forward differences*
 - step 5.2.2 Normalize the value found in step 5.2.1 and insert it into G-Map*

Step 1 involves taking the first derivatives and performing the cross-product. *Step 2* computes the maximum rates of change of κ along u - and along v -directions for the entire surface, i.e. in the range $0 \leq u, v \leq 1$. To find the extrema of bivariate functions over bounded domains, a standard numerical solution can be applied [Press 86]. The results are used in *step 3* to compute the intervals of sampling, i.e. δu dictates the difference in the parameter u between consecutive sample curves and δv dictates the increment in parameter v from one sample point to the next along each curve. In *step 4*, the surface is approximated by a series of constant parameter curves. In *step 5*, forward differencing is employed to perform the computation of normals at sample points at constant parametric intervals.

While sampling at uniform intervals in parametric space facilitates efficient computations of sample normals by employing the forward differencing method, the sample normals are not uniformly distributed on the G-Map. Furthermore, the above procedure uses the global maxima (κ_{uMax} and κ_{vMax}) to ensure that the largest gap is within tolerance. This inevitably results in large number of sample normals being generated. It is desirable for the sample size to be reduced for efficiency of subsequent operations. (In this paper, the determination of the G-Map is not an end in itself - the G-Map is to provide a means for the determination of the visibility map. It will be shown later that in the determination of the visibility map only the surface normals at the boundary of the G-Map matter.) As a consequence, those sample normals that do not lie on the boundary of the G-Map may be discarded. A means of achieving this is sought.

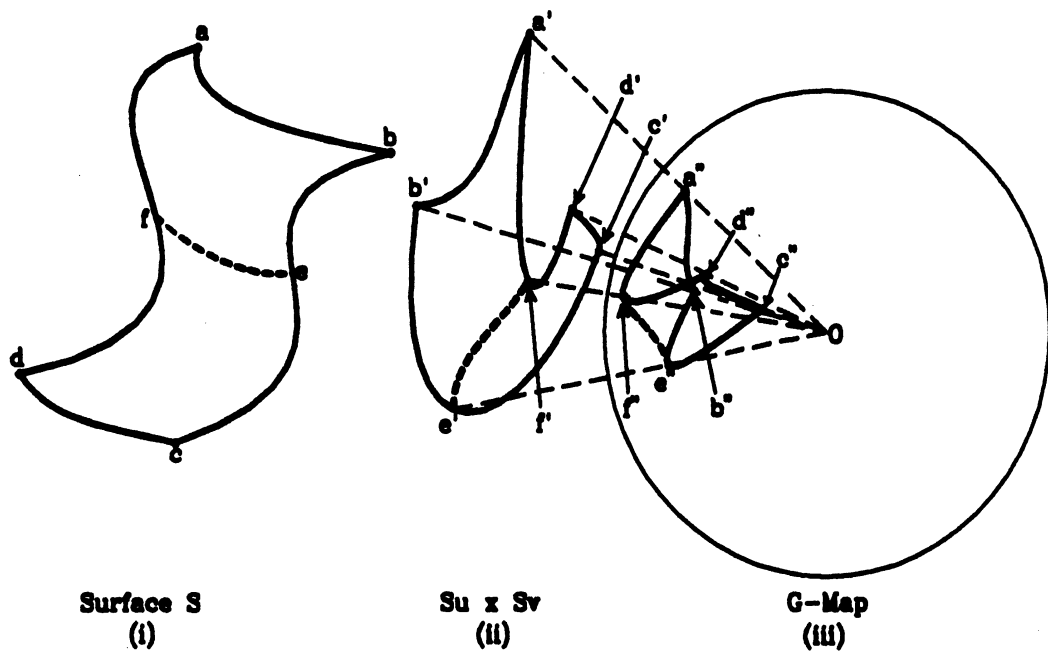


Figure A-1 Traces of a Surface and Traces of Its G-Map

A G-Map that "folds", resulting in part of the boundary of the G-Map being formed from points in the interior of the surface, is illustrated with the help of Figure A-1. $abcd$ is the boundary of a curved surface S in the Cartesian space. Suppose the curve ef on S is a *parabolic curve*, i.e. it consists of *parabolic points*. A parabolic curve for a curved surface is analogous to a point of inflection on a curve, it has the property that all points immediately to one side of it are convex, while all points immediately to the other side are concave. Since the G-Map of a curve in the convex region of a surface is in the same direction as the curve whereas the G-Map of a curve in the concave region is in the opposite direction to that of the curve, the G-Map folds at the parabolic curve. The folding edge $e'f'$ on the G-Map is therefore due to the parabolic curve ef on the surface S .

Instead of computing the parabolic curve directly, the following observation facilitates a simple approach to identifying the sample points in the vicinity of the parabolic curve. The map of $S_u \times S_v$, the "un-normalized G-Map", is a surface in 3-D space. The G-Map is the projection of the $S_u \times S_v$ surface on the unit sphere. For example in Figure A-1, the $S_u \times S_v$ surface of S is shown in (ii), and the G-Map is shown in (iii). From geometric consideration, it can be seen the surface normal vectors on the $S_u \times S_v$ surface for points corresponding to the parabolic curve are orthogonal to the corresponding $S_u \times S_v$ vectors. This condition can be identified by their dot product, i.e.

$$f(u,v) = (S_u \times S_v) \cdot ((S_u \times S_v)_u \times (S_u \times S_v)_v) = 0$$

where $(S_u \times S_v)_u$ and $(S_u \times S_v)_v$ are the first derivatives of $(S_u \times S_v)$ with respect to the parameter u and the parameter v respectively. $f(u,v)$ is a polynomial in parameters u and v if S is polynomial, and if S is a rational surface similar analysis may be used by considering only the numerator of $f(u,v)$ which is polynomial. A standard numerical method may be used to solve the above equation for sets of u and v values where the above equation holds. Alternatively, the forward differencing method may also be employed to compute the values of $f(u,v)$ at each of the sample point at the same time as the computation of $(S_u \times S_v)$. Note that if the values of $f(u,v)$ at two consecutive sample points have different signs, then a point on the folding edge lies between these two consecutive sample points. This observation provides a means of deciding which of the sample normals should be retained, and which may be discarded. The modified algorithm, *Procedure G-Map**, which generates only those normals that have probability of being on or close to the "boundary" of the G-Map, is summarized:

Procedure G-Map*(S)

- step 1** Determine the coefficients of $(S_u \times S_v)$
- step 2** For $u=0$ curve Do
- step 2.1** Compute K_{vMax}
- step 2.2** $\delta v = tol / K_{vMax}$
- step 2.3** Compute initial forward differences for $(S_u \times S_v)$ based on δv increment
- step 2.4** For $v := 0$ To 1 in step of δv Do
- step 2.4.1** Determine value of $(S_u \times S_v)$ by forward differencing
- step 2.4.2** Normalize the value found in step 2.4.1 and insert it into G-Map
- step 3** Repeat step 2 for $u=1$ curve
- step 4** Compute K_{uMax} for entire surface
- step 5** $\delta u = tol / K_{uMax}$
- step 6** Approximate $(S_u \times S_v)$ by a series of constant- u curves spaced δu apart
- step 7** For each of the constant- u curves (excl $u=0$ and $u=1$ curves) Do
- step 7.1** Compute K_{vMax}
- step 7.2** $\delta v = tol / K_{vMax}$
- step 7.3** Compute initial forward differences for $(S_u \times S_v)$ based on δv increment
- step 7.4** Determine value of $(S_u \times S_v)$ for $v=0$
- step 7.5** Normalize the value found in step 7.4 and insert it into G-Map
- step 7.6** Compute initial forward differences for $f(u,v)$ based on δv increment
- step 7.7** Determine value of $f(u,v)$ for $v=0$
- step 7.8** For each increment of v by δv Do
- step 7.8.1** Determine new value of $(S_u \times S_v)$ by adding the relevant forward differences
- step 7.8.2** Determine new value of $f(u,v)$ by adding the relevant forward differences
- step 7.8.3** If $f(u,v)$ changes sign Then
- Normalize the value found in step 7.8.1 and insert it into G-Map
- step 7.9** If $f(u,v)$ does not changes sign Then
- Normalize the value found in step 7.8.1 and insert it into G-Map

In *Procedure G-Map**, *Step 2* and *step 3* compute and store all the sample normals along the two boundary curves corresponding to $u=0$ and $u=1$. The sample normals are generated at intervals which satisfy the gap tolerance along the corresponding curve only. *Step 4*, *step 5* and *step 6* compute the maximum rate of change of κ along u -direction for the entire surface, i.e. in the range $0 \leq u, v \leq 1$, and use it to compute the interval δu between sample curves. In *step 7*, only the sample normals that are from points on the boundaries of the progenitor surface as well as those points that have the probabilities of being on, or in the vicinity of, the boundary of the G-Map, are accepted. In this way, only a small fraction of the sample normals generated by the previous *Procedure G-Map* will be generated by *Procedure G-Map**. In *step 7.5*, the sample normal at the beginning of each curve is accepted into the G-Map. In *step 7.8.3*, the sample normal in the vicinity of a parabolic point is accepted into the G-Map. *Step 7.9* ensures that the normal at the end of each sample curve is accepted into the G-Map.

APPENDIX B

Determining Spherical Convex Hull

Convex hull algorithms for a set of points lying on a plane are well researched [Preparata 85, Sedgewick 83]. If the set of points P in S^2 are hemispherical, i.e. they can be contained within a hemisphere, then it is possible to map P into a set of points P' in the plane by central projection. (If P is not hemispherical, then central projection onto two parallel planes suffice. See [Chen 90] for details.) Since the central projection is a geodesic projection, the great arcs are projected as straight lines which form the edges of the convex hull in the plane corresponds to the great arcs in the spherical convex hull. Figure B-1 illustrates the projection of the spherical convex hull $abcd$ onto a plane as a planar convex hull $a'd'c'b'$. *Procedure SCH* below outline the steps involved in finding the spherical convex hull via central projection.

Procedure SCH(G-Map)

- step 1: Centrally Project the points in the G-Map onto a plane*
- step 2: Determine the convex hull in the projected plane*
- step 3: Reverse the projection to get the spherical convex hull*

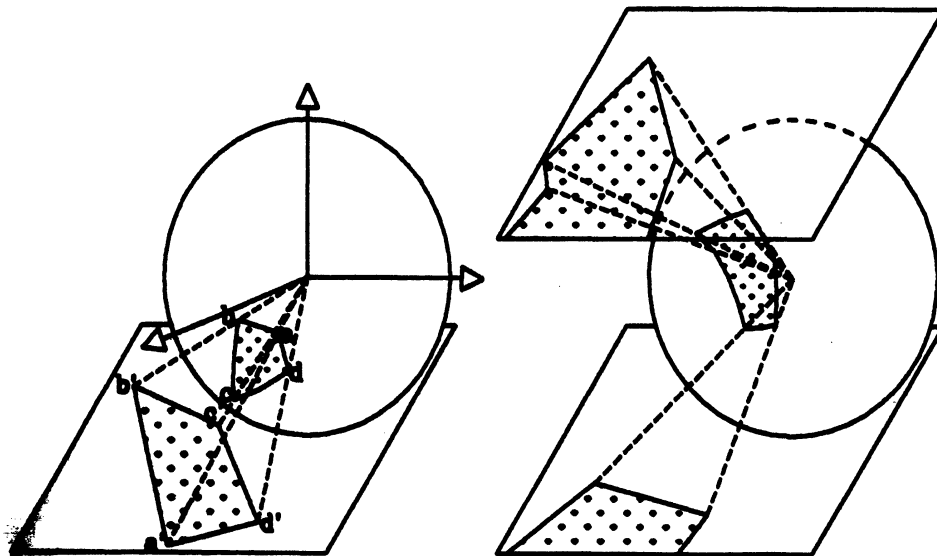


Figure B-1 Projection of Spherical Convex Hull

Step 1 can be performed in time $O(n)$ where n is the number of given points. *Step 2* can be performed by using Graham Scan which is $O(n \log n)$ [Preparata 85]. *Step 4* is the reverse projection and it can be done in time $O(n)$. Therefore, the overall complexity of *Procedure SCH* is $O(n \log n)$.

REFERENCES

- Anderson DP: "Efficient Algorithms for Automatic Viewer Orientation". *Comp & Grap*, Vol. 9, No. 4, p407-413, 1985.
- Besl PJ: *Surfaces in Range Image Understanding*, Springer-Verlag, NY, 1988.
- Bollobas B: *Graph Theory, An Introductory Course*. Springer-Verlag, NY, 1979.
- Calladine CR: "Gaussian Curvature and Shell Structure". In Gregory JA (ed) *The Mathematics of Surfaces*, Clarendon Press, Oxford, 1986.
- Chen YJ and Ravani B: "Offset Surface Generation and Contouring in Computer Aided Design", *ASME J. Mech. Transm. and Auto. in Design*, Sep 1987.
- Chen LL and Woo TC: "*Computational Geometry on the Sphere with Application to Automated Machining*". to appear in *ASME Trans J of Mechanical Design*. Also available as Technical Report 89-30, IOE Dept, U of Michigan, 1989.
- do Carmo MP: *Differential Geometry of Curves and Surfaces*. Prentice-Hall, NJ, 1976.
- Farouki RT: "The approximation of non-degenerate surfaces", *CAGD*, Vol. 2, No. 1, p257-279, 1985.
- Faux ID and Pratt MJ: *Computational Geometry for Design and Manufacture*. Halsted Press, NY, 1979.
- Foley JD and Van Dam A: *Fundamentals of Interactive Computer Graphics*. Addison-Wesley, Reading, MA, 1982.
- Gan J: *Spherical Algorithms for Setup Orientations of Workpieces with Sculptured Surfaces*. Ph.D. Dissertation, IOE Dept, U of Michigan, 1989.
- Gerald CF and Wheatley PO: *Applied Numerical Analysis*. Addison-Wesley, Reading, Massachusetts, 1984.
- Hilbert D and Cohn-Vossen S: *Geometry and the Imagination*. Translated by P Nementi. Chelsea, NY, 1983.

- Kim DS: *Cones on Bezier Curves and Surfaces*. Ph.D. Dissertation, IOE Dept, U. of Michigan, 1990.
- Laugwitz D: *Differential and Riemannian Geometry*. translated by Steinhardt F. Academic Press, NY, 1965.
- Preparata FP and Shamos MI: *Computational Geometry - An Introduction*. Springer-Verlag, NY, 1985.
- Press WH, Flannery BP, Teukolsky SA and Vetterling WT: *Numerical Recipes - The Art of Scientific Computing*. Cambridge, 1986.
- Sedgewick R: *Algorithms*. Addison Wesley, Reading, Massachusetts, 1983.
- Shamos MI and Hoey D: "Closest-point Problems". In *16th Annual Symposium on Foundations of Computer Science*. p155-162, 1975.
- Spyridi AJ and Requicha AAG: "Accessibility Analysis for the Automatic Inspection of Parts". Proc. Int'l. Conf. on Robotics and Automation, Cincinnati, Oh, p 1284-1289. May 13-18, 1990.
- Sripradisvarakul T and Jain R: "Generating Aspect Graph for Curved Objects". Proc. Workshop on Interpretation of 3D Scenes, Austin, Tx, p 109-115. Nov 27-29, 1989.
- Stanton RG: *Numerical Methods for Science and Engineering*. Prentice-Hall, NJ, 1960.
- Swamy MNS and Thulasiraman K: *Graphs, Networks, and Algorithms*. John Wiley, NY, 1981.
- Tang K, TC Woo and JG Gan: "Maximum Intersection of Spherical Polygons and Workpiece Orientation for 4- and 5-axis Machining". to appear in *ASME Trans. J. Mechanical Design*. Also available as *Technical Report 89-34, IOE Dept, U of Michigan, Dec 1989*.