Representations of Three-Dimensional Structures 9

9.1 SOLIDS AND THEIR REPRESENTATION

We consider three general classes of representations for rigid solids:

1. Surface or boundary
2. Sweep (in general, generalized cylinders)
3. Volumetric (in general, constructive solid geometry)

The semantics of solid representations is intuitively clear but sometimes mathematically tricky. The representations have different computational properties, and readers should keep this in mind when assessing a representation for possible use. As a simple example, a surface representation can describe how an object looks; a volumetric version, which expresses the solid as a combination of subparts, may not explicitly contain information about the surface of the object. However, the solid representation may be better for matching, if it can be structured to reflect functional subparts.

Certainly we believe, as do others, that model-based vision will ultimately have to confront the issues of geometric modeling in three dimensions [Nishihara 1979]. Ultimately, nonrigid as well as rigid solids will have to be represented. The characterization of nonrigid solids presents very challenging problems.

Nonrigid solids are often a useful way to model time-varying aspects of objects. Here, again, the kind of model that is best depends heavily on the domain. For example, a useful mammal model may be one with a piecewise rigid linkage (for the skeleton) and some elastic covering (for the flesh). Computer vision in the domain of mammals, either static in various positions or actually moving, might be based on generalized cylinders (Section 9.3). However, another nonrigid domain is that of heart chambers, that change through time as the heart beats. Here the skeleton is a much less intuitive notion, so a different model of nonrigidity may apply. In most cases, nonrigid objects are modeled as parameterized rigid objects. In the example of the human figure, the parameters may be joint angles for linkages representing the skeleton.

The last part of this chapter deals with understanding line drawings, an influential and well-publicized subfield of computer vision. This seemingly simple and accessible domain avoids many of the problems involving early processing and segmentation, yet it is important because it has furnished several important algorithmic and geometric insights. An important breakthrough in this domain was a move from "image understanding" in two dimensions to to an approach based on the three-dimensional world and laws governing three-dimensional solids.

9.2 SURFACE REPRESENTATIONS

The *enclosing surface*, or *boundary*, of a well-behaved three-dimensional object should unambiguously specify the object [Requicha 1980]. Since surfaces are what is seen, these representations are important for computer vision. Section 9.2.1 considers mainly planar polyhedral surface representations. More complex "sculptured surfaces" [Forrest 1972; Barnhill and Riesenfeld 1974; Barnhill 1977] are treated in Section 9.2.2. Some useful surfaces are defined as functions of three-dimensional directions from a central point of origin. Two of these are mentioned in Section 9.2.3.

9.2.1 Surfaces with Faces

Figure 9.1 shows the solid representation scheme most familiar to computer scientists. Solids are represented by their boundaries, or enclosing surfaces, which are represented in terms of such primitive entities as unbounded mathematical surfaces, curves, and points which together may be used to define "faces."

In general, a boundary is made up of a number of faces; faces are represented by mathematical surfaces and by information about their own boundaries (consisting of edges and possibly vertices). A closed surface such as the sphere or a spherical harmonic surface of Section 9.2.3 may be thought of as having only one face.

To specify a boundary representation, one must answer several important questions of representation design. What is a face, and how are faces represented? What is an edge, and how are edges represented? How much extra information (i.e., useful but redundant relationships and geometric data) should be kept?

What is a face? "Face" is an initially appealing but imprecise notion; it is at its clearest in the context of planar polyhedra. A face should probably always be a subset of the boundary of an object; presumably, it should have area but no dangling edges or isolated points, and the union of all the faces should make up the boundary or the object. Beyond this little can be said. For many purposes it makes sense to have faces overlap; it may be elegant to consider the letter on an alphabet block a special kind of face on the block that is a subset of the face making up the side of the block. On the other hand, it is easy to imagine applications in which faces should not overlap in area (then one easily can compute the surface area of a solid from its faces). In some objects, just what the faces are is purely a matter of

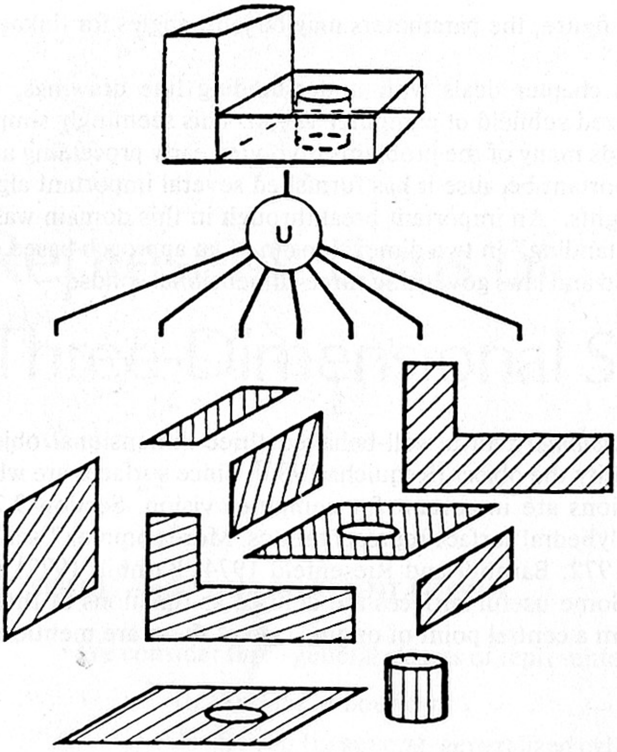


Fig. 9.1 A volume and the faces of a boundary representation.

opinion (Fig. 9.2). In short, any single definition of face is likely to be inadequate for some important application.

The availability of explicit representations of edges, faces, and vertices makes boundary representations quite useful in computer vision and graphics. The computational advantages of polyhedral surfaces are so great that they are often pressed into service as approximate representations of nonpolyhedra (Fig. 9.3).

An influential system for using face-based representations for planar polyhedral objects is the "winged edge" representation [Baumgart 1972]. Included in the system is an editor for creating complex polyhedral objects (such as that of Fig. 9.3) interactively. The system uses rules for construction based on the theorem of Euler that if V is the number of vertices in a polyhedron, E the number of edges, and F the number of faces, then $V - E + F = 2$. In fact, the formula can be extended to deal with non-simply connected bodies. The extended relation is $V - E + F = 2(B - H)$, with B being the number of bodies and H being the

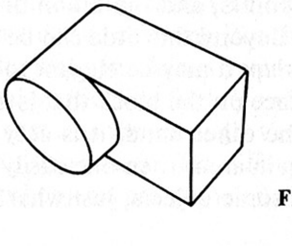


Fig. 9.2 What are the faces?

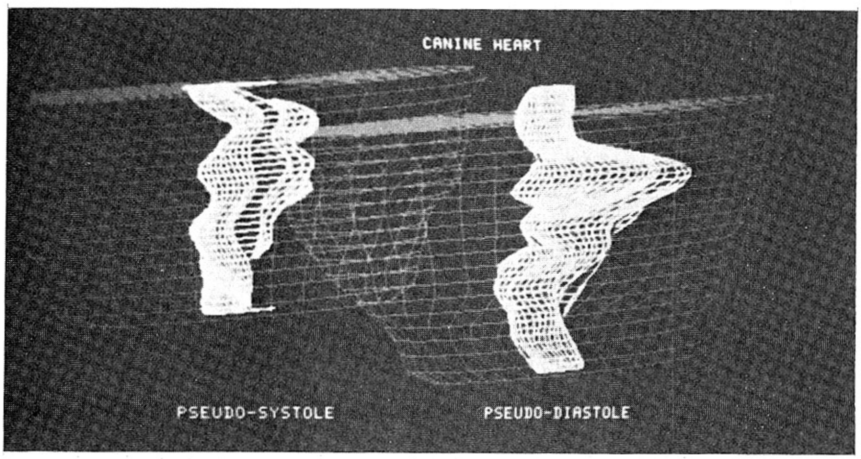


Fig. 9.3 A polyhedral approximation to a portion of a canine heart at systole and diastole. Both exterior (coarse grid) and interior surfaces (fine grid) are shown.

number of holes, or "handles," each resulting from a hole through a body [Lakatos 1976]. Baumgart's system uses these rules to oversee and check certain validity conditions on the constructions made by the editor.

The "winged edge" polyhedron representation achieves many desiderata for boundary representations in an elegant way. This representation is presented below to give a flavor of the features that have been traditionally found useful. Given as primitives the vertices, edges, faces, and polyhedra themselves, and given various relations between these primitives, one is naturally thinks of a record and pointer (relational) structure in which the pointers capture the binary relations and the records represent primitives and contain data about their locations or parameters.

In the winged edge representation, there are data structure records, or nodes, which contain fields holding data or links (pointers) to other nodes. An example using this structure to describe a tetrahedron is shown in Fig. 9.4. There are four kinds of nodes: vertices, edges, faces, and bodies. To allow convenient access to these nodes, they are arranged in a circular doubly linked list. The body nodes are actually the heads of circular structures for the faces, edges, and vertices of the body. Each face points to one of its perimeter edges, and each vertex points to one of the edges impinging on it. Each edge node has links to the faces on each side of it, and the vertices at either end.

Figure 9.4 shows only the last-mentioned links associated with each edge node. The reader may notice the similarity of this data structure with the data structure for region merging in Section 5.4. They are topologically equivalent. Each edge also has associated four links which give the name "winged edge" to the representation. These links specify neighboring edges in order around the two faces which are associated with the edge. The complete link set for an edge is shown in Fig. 9.5, together with the link information for bodies, vertices, and faces. To allow unambiguous traversal around faces, and to preserve the notion of

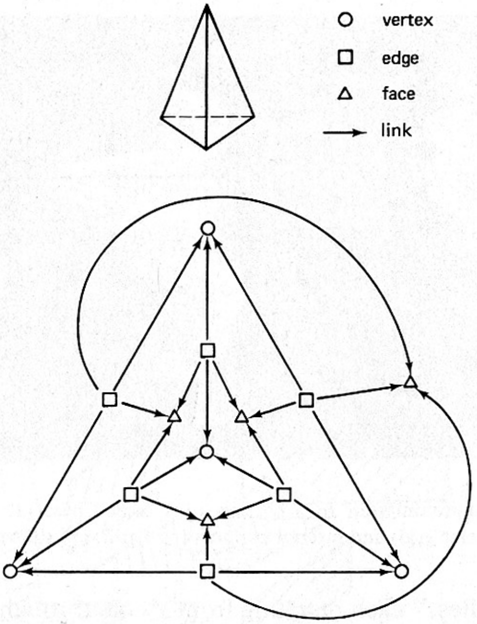


Fig. 9.4 A subset of edge links for a tetrahedron using the "winged edge" representation.

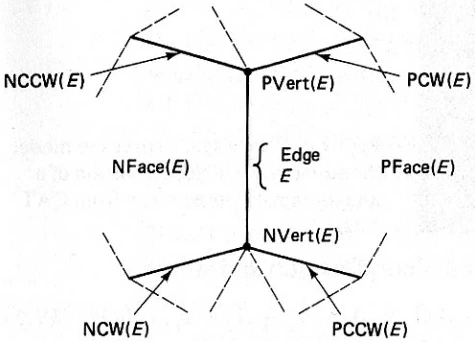
interior and exterior of a polyhedron, a preferential ordering of vertices and lines is picked (counterclockwise, say, as seen from outside the polyhedron).

Data fields in each vertex allow storage of three-dimensional world coordinates, and also of three-dimensional perspective coordinates for display. Each node has fields specifying its node type, hidden line elimination information, and other general information. Faces have fields for surface normal vector information, surface reflectance, and color characteristics. Body nodes carry links to relate them to a tree structure of bodies in a scene, allowing for hierarchical arrangement of subbodies into complex bodies. Thus body node data describe the scene structure; face node data describe surface characteristics; edge node data give the topological information needed to relate faces, edges, and vertices; and vertex node data describe the three-dimensional vertex location.

This rich and redundant structure lends itself to efficient calculation of useful functions involving these bodies. For instance, one can easily follow pointers to extract the list of points around a face, faces around a point, or lines around a face. Winged edges are not a universal boundary representation for polyhedra, but they do give an idea of the components to a representation that are likely to be useful. Such a representation can be made efficient for accessing all faces, edges, or vertices; for accessing vertex or edge perimeters; for polyhedron building; and for splitting edges and faces (useful in construction and hidden-line picture production, for instance).

9.2.2 Surfaces Based on Splines

The natural extension of polyhedral surfaces is to allow the surfaces to be curved. However, with an arbitrary number of edges for the surface, the interpolation of



(a)

Boundary Representation Node Accessing Functions

1. To enter and traverse Face ring of a body:
   NextFace, PreviousFace: Body or Face → Face
2. To enter and traverse Edge ring of a body:
   NextEdge, PreviousEdge: Body or Edge → Edge
3. To enter and traverse Vertex ring of a body:
   NextVert, PreviousVert: Body or Vertex → Vertex
4. First Edge of a Face:
   FirstEdge: Face → Edge
5. FirstEdge of a Vertex:
   FirstEdge: Vertex → Edge
6. Faces of an Edge: [see diagram in (a)]
   N(ext) Face, P(revious) Face: Edge → Face
7. Vertices of an Edge: [see diagram in (a)]
   N(extVert, P(revious) Vert: Edge → Vertex
8. Neighboring Wing Edges of an Edge: [see diagram in (a)]
   NCW, NCCW: Edge → Edge (NFace Edge Clockwise,
   NFace Edge Counterclockwise)

   PCW, PCCW: Edge → Edge (PFace Edge Clockwise,
   PFace Edge Counterclockwise

(b)

Fig. 9.5 (a) Node accessing functions. (b) Semantics of winged edge functions.

interior face points becomes impractically complex. For that reason, the number of edges for a curved face is usually restricted to three or four.

A general technique for approximating surfaces with four-sided surface *patches* is that of Coons [Coons 1974]. Coons specifies the four sides of the patch with polynomials. These polynomials are used to interpolate interior points. Although this is appropriate for synthesis, it is not so easy to use for analysis. This is because of the difficulty of registering the patch edges with image data. A given surface will admit to many patch decompositions.

An attractive representation for patches is splines (Fig. 9.6). In general, two-dimensional spline interpolation is complex: For two parameters *u* and *v* interpolate with

$$\mathbf{x}(u, v) = \sum_i \sum_j V_{ij} B_{ij}(u, v) \tag{9.1}$$

similar to Eq. (8.4). However, for certain applications a further simplification can be made. In a manner analogous to (8.9) define a grid of knot points $\mathbf{v}_{ij}$ corresponding to $\mathbf{x}_{ij}$ and related by

$$\mathbf{x}_{ij} = M\mathbf{v}_{ij} \tag{9.2}$$

Now rather than interpolating in two dimensions simultaneously, interpolate in one direction, say *t*, to obtain

$$\mathbf{x}_{ij}(t) = [t^3 \quad t^2 \quad t \quad 1][C][\mathbf{v}_{i-1,j_0}, \mathbf{v}_{i,j_0}, \mathbf{v}_{i+1,j_0}, \mathbf{v}_{i+2,j_0}]^T \tag{9.3}$$

for each value of *j*. Now compute $\mathbf{v}_{ij}(t)$ by solving

$$\mathbf{x}_{ij}(t) = M\mathbf{v}_{ij}(t) \tag{9.4}$$

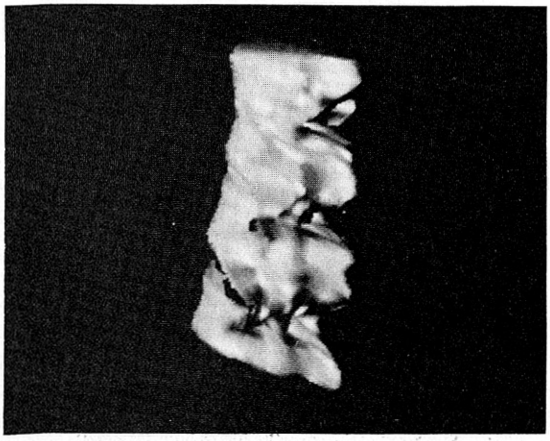


Fig. 9.6 Using spline curves to model the surface of an object: a portion of a human spinal column taken from CAT data.

for each value of *t*. Finally, interpolate in the other direction and solve:

$$\mathbf{x}_{ij}(s,\ t) = [s^3 \quad s^2 \quad s \quad 1][C][\mathbf{v}_{i-1,j}(t),\ \mathbf{v}_{i,j}(t),\ \mathbf{v}_{i+1,j}(t),\ v_{i+2,j}(t)] \qquad (9.5)$$

This is the basis for the spline filtering algorithm discussed in Section 3.2.3.

Some advantages of spline surfaces for vision are the following.

1. The spline representation is economical: the space curves are represented as a sparse set of knot points from which the underlying curves can be interpolated.
2. It is easy to define splines interactively by giving the knot points; reference representations may be built up easily.
3. It is often useful to search the image in a direction perpendicular to the model reference surface. This direction is a simple function of the local knot points.

9.2.3 Surfaces That Are Functions on the Sphere

Some surfaces can be expressed as functions on the "Gaussian sphere." (the distance from the origin to a point on the surface is a function of the direction of the point, or of its longitude and latitude if it were radially projected on a sphere with the center at the origin.) This class of surfaces, although restricted, is useful in some application areas [Schudy and Ballard 1978, 1979]. This section explores briefly two schemes for representation of these surfaces. The first specifies explicitly the distance of the surface from the origin for a set of vector directions from the origin. The second is akin to Fourier descriptors; an economically specified set of coefficients characterizes the surface with greater accuracy as the number of coefficients increases.

Direction–Magnitude Sets

One approximation to a spherical function is to specify a number of three-dimensional direction vectors from the origin and for each a magnitude. This is equivalent to specifying a set of $(\theta,\ \phi,\ \rho)$ points in a spherical coordinate system (Appendix 1). These points are on the surface to be represented; connecting them yields an approximation.

It is often convenient to represent directions as points on the unit (Gaussian) sphere centered on the origin. The points may be connected by straight lines to form a polyhedron with triangular, hexagonal or rhomboidal faces. Moving the points on the sphere out (or in) by their associated magnitude distorts this polyhedron, moving its vertices radically out or in.

The spherical function determines the distance of face vertices from the origin. Resolution at the surface increases with the number of faces. An approximately isotropic distribution of directions over the surface may be obtained by placing the face vertices (directions) in accordance with "geodesic dome"–like calculations which make the faces approximately equilateral triangles [Clinton 1971].

Although the geodesic tesselation of the sphere's surface is more complex than a straightforward (latitude and longitude, say) division, its pleasant properties of isotropy and display [Brown 1979a; 1979b; Schudy and Ballard 1978] sometimes recommend it. Some example shapes indicating the range of representable surfaces are given in Fig. 9.7. Methods for tesselating the sphere are given in Appendix 1.

Spherical Harmonic Surfaces

In two dimensions, Fourier coefficients can give approximations to certain curved boundaries (Section 8.3.4). Analogously in three dimensions, a set of orthogonal functions may be used to express a closed boundary as a set of coefficients when the boundary is a function on the sphere. One such decomposition is *spherical harmonics.* Low order coefficients capture gross shape characteristics; higher order coefficients represent surface shape variations of higher spatial frequency. The function with $m = 0$ is a sphere, the three with $m = 1$ represent translation about the origin, the five with $m = 2$ are similar to prolate and oblate spheroids, and so forth, the lobedness of the surfaces increasing with m. A sample three dimensional shape and its "description" is shown in Fig. 9.8.

Spherical harmonics are analogs on the sphere of Fourier functions on the plane; like Fourier functions, they are smooth and continuous to every order. They may be parameterized by two numbers, m and n; thus they are a doubly infinite set of functions which are continuous, orthogonal, single-valued, and complete on the

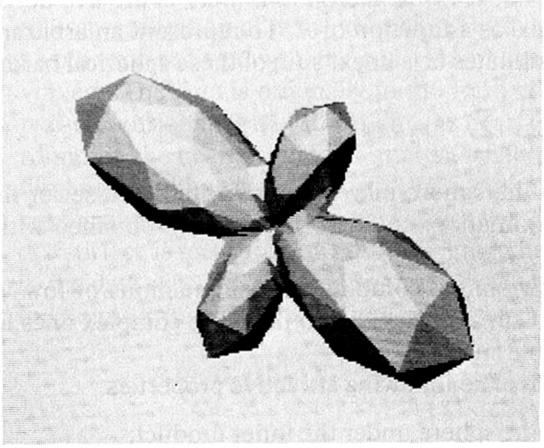


Fig. 9.7 Sample surfaces described by some 320 triangular facets in a geodesic tesselation.

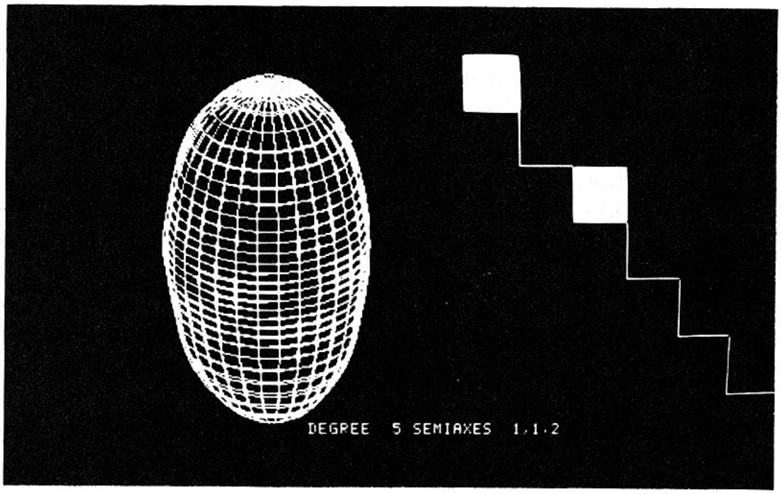


Fig. 9.8 A spherical harmonic function description of an ellipsoid. Coefficients are displayed on the right as grey levels in the matrix format

$$\begin{array}{lllll}
u_{00} & & & & \\
 & u_{01} & v_{11} & & \\
 & u_{11} & u_{02} & v_{12} & v_{22} \\
 & & u_{12} & & \\
 & & u_{21} & &
\end{array}$$

sphere. In combination, the harmonics can thus produce all "well-behaved" spherical functions.

The spherical harmonic functions $U_{mn}\ (\theta, \phi)$ and $V_{mn}\ (\theta, \phi)$ are defined in polar coordinates by:

$$U_{mn}(\theta, \phi) = \cos(n\theta)\sin^n(\phi)P(m, n, \cos(\phi)) \tag{9.6}$$

$$V_{mn}(\theta, \phi) = \sin(n\theta)\sin^n(\phi)P(m, n, \cos(\phi)) \tag{9.7}$$

with $m = 0, 1, 2, ..., M$; $n = 0, 1, ..., m$. Here $P(m, n, x)$ is the nth derivative of the mth Legendre polynomial as a function of x. To represent an arbitrary shape, let the radius R in polar coordinates be a linear sum of these spherical harmonics:

$$R(\theta, \phi) = \sum_{m=0}^{M} \sum_{n=0}^{m} A_{mn}U_{mn}(\theta, \phi) + B_{mn}V_{mn}(\theta, \phi) \tag{9.8}$$

Any continuous surface on the sphere may be represented by a set of these real constants; reasonable approximations to heart volumes are obtained with $m \leqslant 5$ [Schudy and Ballard 1979].

Figure 9.9 shows a few simple combinations of functions of low values of (m, n). The sphere, or $(0, 0)$ surface, is added to the more complex ones to ensure positive volumes and drawable surfaces.

Spherical harmonics have the following attractive properties.

1. They are orthogonal on the sphere under the inner product;

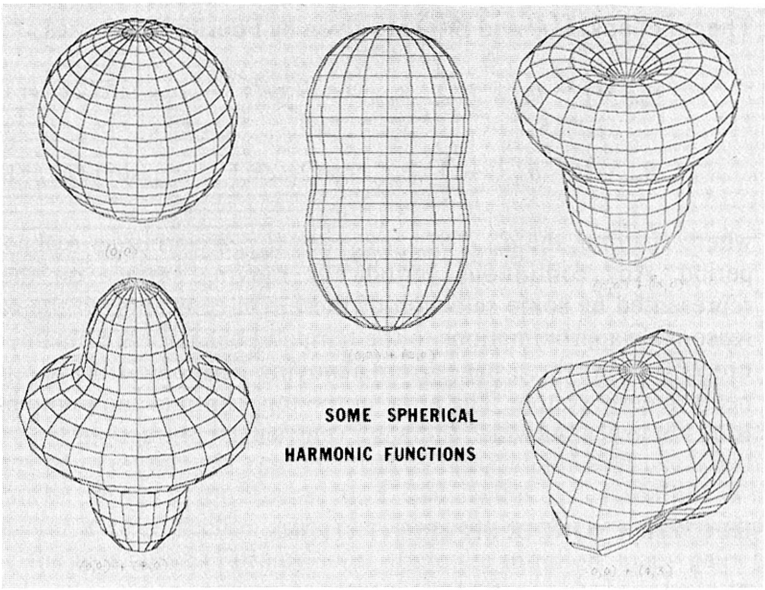


Fig. 9.9 Simple combinations of functions.

$$(u, v) = \int uv \sin\phi \, d\theta \, d\phi$$

2. The functions are arranged in increasing order of spatial complexity.
3. The whole set is complete; any twice-differentiable function on the sphere can be approximated arbitrarily closely.

Spherical harmonics can provide compact, nonredundant descriptions of surfaces that are useful for analysis of shape, but are less useful for synthesis. The principal disadvantages are that the primitive functions are not necessarily related to the desired final shape in an intuitive way, and changing a single coefficient affects the entire resulting surface.

An example of the use of spherical harmonics as a volume representation is the representation of heart volume [Schudy and Ballard 1978, 1979]. In extracting a volume associated with the heart from ultrasound data, a large mass of data is involved. The data is originally in the form of echo measurements taken in a set of two-dimensional planes through the heart. The task is to choose a surface surrounding the heart volume of interest by optimization techniques that will fit three dimensional time-varying data. The optimization involved is to find the best coefficients for the spherical harmonics that define the surface. The goodness of fit of a surface is measured by how well it matches the edge of the volume as it appears in the data slices. To extend spherical harmonics to time-varying periodic data, let the radius R in polar coordinates be a linear sum of these spherical harmonics:

$$R(\theta, \phi, t) = \sum_{m=0}^{M} \sum_{n=0}^{m} A_{mn}(t)\, U_{mn}(\theta, \phi) + B_{mn}(t)\, V_{mn}(\theta, \phi) \qquad (9.9)$$

The functions $A(t)$ and $B(t)$ are given by Fourier time series:

$$A_{mn}(t) = a_{mno} + \sum_{i=1}^{I} a_{mni} \cos(2\pi t/\tau) + b_{mni} \sin(2\pi t/\tau) \tag{9.10}$$

$$B_{mn}(t) = b_{mno} + \sum_{i=1}^{I} c_{mni} \cos(2\pi t/\tau) + d_{mni} \sin(2\pi t/\tau) \tag{9.11}$$

where t is time, the a_{mni}, b_{mni}, c_{mni}, and d_{mni} are arbitrary real constants, and τ the period. Any continuous periodically moving surface on the sphere may be represented by some selection of these real constants; in the cardiac application, reasonable approximations to the temporal behavior are obtained with $t \leqslant 3$. Figure 9.10 shows three stages from a moving-harmonic-surface representation of the heart in early systole. The atria, at the top, contract and pump blood into the ventricles below, after which there is a ventricular contraction.

9.3 GENERALIZED CYLINDER REPRESENTATIONS

The volume of many biological and manufactured objects is naturally described as the "swept volume" of a two-dimensional set moved along some three-space curve. Figure 9.11 shows a "translational sweep" wherein a solid is represented as the volume swept by a two-dimensional set when it is translated along a line. A "rotational sweep" is similarly defined by rotating the two-dimensional set around an axis. In "three-dimensional sweeps," volumes are swept. In a "general" sweep scheme, the two-dimensional set or volume is swept along an arbitrary space curve, and the set may vary parametrically along the curve [Binford 1971; Soroka and Bajcsy 1976; Soroka 1979a; 1979b; Shani 1980]. General sweeps are quite a popular representation in computer vision, where they go by the name *generalized cylinders* (sometimes "generalized cones").

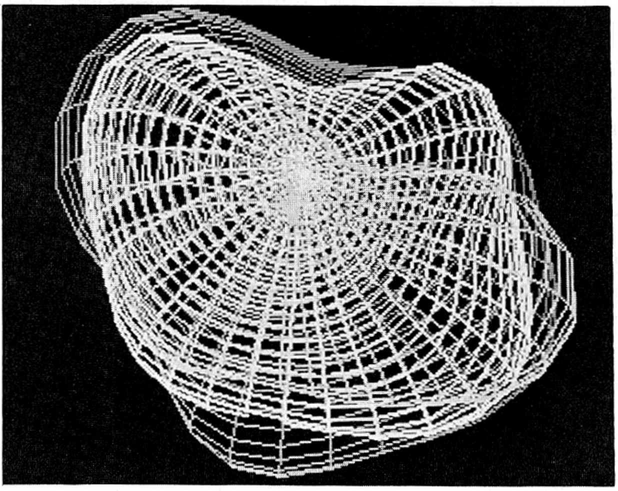


Fig. 9.10 Three stages from a moving harmonic surface (*see text and color insert*).