COMPUTER VISION AND IMAGE UNDERSTANDING
Vol. 61, No. 2, March, pp. 154–170, 1995

Parametrization of Closed Surfaces for 3-D Shape Description

CH. BRECHBÜHLER, G. GERIG, AND O. KÜBLER

Communication Technology Laboratory, Image Science, Swiss Federal Institute of Technology (ETH), CH-8092 Zurich, Switzerland

Received November 19, 1993; accepted August 22, 1994

This paper presents procedures for the explicit parametric representation and global description of surfaces of simply connected 3-D objects. The novel techniques overcome severe limitations of earlier methods (restriction to star-shaped objects (D. H. Ballard and Ch. M. Brown, *Computer Vision*, Prentice–Hall, Englewood Cliffs, NJ, 1981), constraints on positioning and shape of cross-sections (F. Solina and R. Bajcsy, *IEEE Trans. Pattern Anal. Mach. Intell.* 12(2), 1990, 131–147; L. H. Staib and J. S. Duncan, in *Visualization in Biomedical Computing 1992* (R. A. Robb, Ed.), Vol. Proc. SPIE 108, pp. 90–104, 1992), and nonhomogeneous distribution of parameter space). We parametrize the surface by defining a continuous, one-to-one mapping from the surface of the original object to the surface of a unit sphere. The parametrization is formulated as a constrained optimization problem. Practicable starting values are obtained by an initial mapping based on a heat conduction model. The parametrization enables us to expand the object surface into a series of spherical harmonic functions, extending to 3-D the concept of elliptical Fourier descriptors for 2-D closed curves (E. Persoon and K. S. Fu, *IEEE Trans. Syst. Man Cybernetics* 7(3), 1977, 388–397; F. P. Kuhl and Ch. R. Giardina, *Comput. Graphics Image Process.* 18(3), 1982, 236–258). Invariant, object-centered descriptors are obtained by rotating the parameter net and the object into standard positions. The new methods are illustrated with 3-D test objects. Potential applications are recognition, classification, and comparison of convoluted surfaces or parts of surfaces of 3-D shapes. © 1995 Academic Press, Inc.

1. INTRODUCTION

With the proliferation of high-quality volumetric image data, especially for the medical community, and new segmentation methods for multidimensional image data, 3-D objects become available and are ready for structural analysis. Most often, volumetric objects are represented by a binary voxel representation or by a triangulation of the surface. Although these representations make possible a 3-D rendering for visually capturing the object properties, both lack descriptive power as they are based on huge lists of voxels or surface elements. Characterizing and understanding shape properties, however, requires a representation which captures global and local shape features with a small number of parameters. Such a concise description could be useful for a comparison of various objects, for finding dissimilarities, for matching objects to predefined models, and for an efficient reconstruction and manipulation of objects.

A shape descriptor must be general enough to handle very different shapes, but should be capable of accurately representing global as well as local features of objects. Shape analysis favors object-centered volume or surface descriptions, e.g., using polynomials, triangulation meshes, generalized cylinders, medial manifolds, or spherical harmonics. Furthermore, the resulting description should be independent of the object position and orientation.

Some shape description methods based on mesh-like surface models rely on a robust and reproducible surface parametrization in a two-coordinate space. While tracking a contour in 2-D images is easily done, the extension to higher dimensions is nontrivial and requires the development of new concepts. Thus far, representation methods for mapping an object surface onto a sphere have been limited to represent only star-shaped or convex objects, as they start from an initial radial surface function $r(\theta, \phi)$, [1, 6]. Staib and Duncan [3] discuss the use of a parameter space with torus topology, which can be deformed into a tube by squeezing the torus cross-section to a thin ribbon. Closed surfaces are obtained by considering tubes whose ends close up to a point. This approach clearly illustrates some principal difficulties which can also be found in other parametrization techniques.

- The idea of warping a torus to a closed surface poses the problem that the *parameters have different functionalities*. One parameter defines a kind of spine along which cross-sections are stacked up. The choice of the endpoints of this spine decisively determines the solution and even determines whether an object can be parametrized.
- Squeezing a circle to a line (as done with the tubular torus cross-section) results in a *nonhomogeneous distribution of parameters* on the object surface. Although continuous, the representation of a line by harmonic functions results in a clustering of parameters near the edges where

1077-3142/95 $6.00

the tracing direction changes. Furthermore, closing a cylinder at both open ends causes further distortion to the parameter net.

• Warping a torus to a tube and finally to a closed surface shows that the parametrization does *not* result in a *one-to-one mapping* of surface points to parameters. The walls of the tube are traced up and down in order to avoid discontinuities at the open edges, visiting each surface point twice.

Our new approach overcomes these limitations. We present a new method that allows a uniform mapping of an object surface into a two-coordinate space with spherical topology. Our aim is the parametrization of arbitrarily shaped simply connected objects containing intrusions and protrusions. As a mapping of convoluted surface structures onto the surface of a sphere introduces distortions, optimization of the distribution of nodes in parameter space becomes necessary. This problem is solved by nonlinear optimization.

Parametrized surfaces can be expanded into spherical harmonics, hierarchically describing global and local shape properties by spatial frequency constituents [7]. A new method for the generation of descriptors which are invariant to translation, rotation, and scaling is developed. Invariance is crucial for a comparative analysis of different structures.

This paper is organized as follows: Section 2 describes the generation of a relational surface data structure, important both for the initial mapping by a diffusion algorithm and for the optimization step. In Section 3, we present the new parametrization technique including an initial mapping to obtain good starting values and nonlinear optimization. Section 4 discusses the expansion of a 3-D shape into a set of spherical harmonics and the generation of invariant descriptors.

2. SURFACE DATA STRUCTURE

A 3-D binary image containing a single simply 6-connected object represents our input data. We adopt the *cuberille* notion [8]. To enable working with the surface of the object, it must be represented as a data structure that reflects the geometry as well as topological relations. This data structure stores information about all the square faces separating object and background voxels and all the edges and vertices bounding these faces. This initial data structure represents a complete description of the object surface; similar structures are often used for 3-D display purposes [9].

For an n_x by n_y by n_z data volume, the voxel coordinates have the ranges $0 \le x < n_x$, $0 \le y < n_y$, and $0 \le z < n_z$. We interpret each voxel as a cube extending from x to $x + 1$, from y to $y + 1$, and from z to $z + 1$, where (x, y, z) are the nominal coordinates of the voxel. The voxel center is placed at $(x + \frac{1}{2}, y + \frac{1}{2}, z + \frac{1}{2})$ to retain integer coordinates for the corners.

The surface data structure is organized around the vertices. Each surface vertex is listed once in the structure, which gives it a unique identifying (id) number. The entry of a vertex specifies its Cartesian coordinates (x, y, z) and a list of its direct and diagonal neighbors. The neighbors are given in counterclockwise sequence around the vertex when viewed from outside the object; therefore, direct and diagonal neighbors alternate (see Fig. 1).

The following algorithm generates the surface data structure. Object voxels touching the border of the data volume would require a special treatment. Therefore, we make sure that all border voxels belong to the background. We define an *interior vertex* as the common corner of eight voxels of the data volume. In three nested loops—varying the coordinates x from 1 to $n_x - 1$, y from 1 to $n_y - 1$, and z from 1 to $n_z - 1$, respectively—we visit all the interior vertices. Let the z loop be the outermost, so that z varies the slowest and numbers the current plane. If the eight incident voxels of a vertex are homogeneous, i.e., all in the object or all in the background, the vertex is ignored. Otherwise (if the incident voxels are mixed), a new surface vertex has to be entered in the data structure. A surface vertex in plane z may have neighboring surface vertices in the planes $z - 1$, z, and $z + 1$. Therefore, surface vertices must be detected and assigned identifying numbers one plane ahead of time, and the id numbers of the current (z) and previous $(z - 1)$ plane must be kept at hand as well. For each surface vertex in the current plane, the Cartesian coordinates are put into the entry of the vertex. Starting at a 6-neighbor surface vertex, we cycle around the central vertex, going over all incident faces until we are back at the starting neighbor

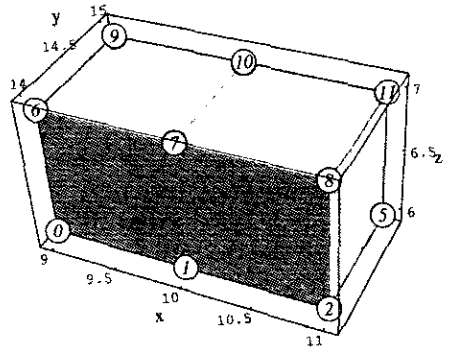

FIG. 1. A two-voxel example object illustrating the surface data structure centered on the vertices. The numbers of the vertices are shown in circles (vertices 3 and 4 are hidden). The data structure entry for a vertex represents its Cartesian coordinates and a list of neighboring nodes. The entry for vertex 7 is {{x = 10, y = 14, z = 7}, *neighbors* = {6, 0, 1, 2, 8, 11, 10, 9}}. Appendix A lists the complete surface data structure of this object.

vertex. The list of the id numbers of all visited direct and diagonal neighbor vertices completes the entry of the central vertex.

Only vertices are stored in the data structure, but information about edges and faces of the surface is also present in the neighbor lists. For instance, vertex 7 in Fig. 1 has the neighbor list {6, 0, 1, 2, 8, 11, 10, 9}. Taking every second number in this list—6, 1, 8, 10—reveals that edges to nodes 6, 1, 8, and 10 emanate from node 7. Overlapping triples of neighbors—{6, 0, 1}, {1, 2, 8}, {8, 11, 10}, and {10, 9, 6} (by wrapping around to the first neighbor)—give the four faces {7, 6, 0, 1}, {7, 1, 2, 8}, {7, 8, 11, 10}, and {7, 10, 9, 6}, all written counterclockwise. In this way, every face of the surface is mentioned four times, once by each corner. Our algorithm requires an additional table that lists every face exactly once. We generate it by visiting all vertices, putting a face in the table only when it is defined from the corner with the smallest id number, i.e., when it is mentioned for the first time.

The data structure generalizes in a very natural way to the case of open surfaces, or surface patches, where vertices on the border of the surface have an odd number of neighbors. All surface vertices that are not part of a border of the surface are called *inner vertices*. Their characteristic is an even number of neighbors.

The possibility of representing surface patches is not exploited here: as the object lies completely within our data volume, we always find a closed surface. All vertices in the list are inner vertices of the surface, and therefore, they all have an even number of neighbors.

The correspondence between the surface net and a graph becomes clear when a vertex is interpreted as a node in the graph, an edge as an arc, and a face as a mesh (four-cycle) [10]. For a simply connected object we get a planar graph with the following topological properties: Four edges bound each face, and each edge bounds two faces and is bounded by two vertices. Depending on the local connectivity, each vertex bounds three to six edges. There are exactly two more vertices than faces; this follows from Euler's relation $n_{\text{vert}} = n_{\text{face}} + 2$.

A surface with its two degrees of freedom is characterized by a polygonal description based on vertex coordinates with three spatial coordinates. Seeking an appropriate parametrization, however, would require a description based on two parameters.

3. SURFACE PARAMETRIZATION

The parametrization, i.e., the embedding of the object surface graph in the surface of the unit sphere, can be posed as a constrained optimization problem. The following paragraphs define the meaning of variables, objective (goal function), constraints, and starting values.

The *variables* of the optimization problem are the coordinates of all vertices. Using two (e.g., spherical) coordinates per vertex would be the most economic representation with respect to storage space, but it would make the equal treatment of all spatial directions difficult and pose the problem of discontinuity and singularities in the parameter space. Therefore, we prefer Cartesian coordinates (u, v, w) for representing a location on the sphere, introducing one virtual degree of freedom per vertex.

There are three kinds of *constraints* on the variables. First, the Euclidean norm of the coordinates of any vertex must be 1. This constraint compensates for the virtual degree of freedom and forces every vertex to lie on the unit sphere in parameter space. Second, we ask for area preservation, which in our context means that any object surface region must map to a region of proportional area on the sphere. To satisfy this requirement, we include one constraint for each spherical quadrilateral (see Fig. 2) that corresponds to an elementary face: its area must be 4π divided by the total number of faces. Third, no angle of any spherical quadrilateral must become negative or exceed π. In contrast to the first two kinds of constraints, these are inequalities. They can be stated as $\sin \alpha_i \geq 0$, $\forall_i$. This adds four inequality constraints for each face.

The *objective* is to minimize the distortion of the surface net in the mapping. It is conceptually similar to angle preservation, and it must tend to make the shape of all the mapped faces as similar to their original square form as possible. To fulfill this goal perfectly, a facet should map to a "spherical square" (see Fig. 2). This can never be reached exactly for all faces except when the object has a very special form, e.g., consists of one single voxel. There are several ways to trade off between the distortions made at different vertices. We observe that the ideal shape of any face, a spherical square, minimizes circumference $\Sigma_{i=0}^{3} s_i$ of any spherical quadrilateral with a given area. At the same time it maximizes $\Sigma_{i=0}^{3} \cos s_i$. These two measures are similar, but not equivalent if summed over the whole net, as they trade off among distortions differently. The second measure punishes too long sides more and honors too short sides less than the first measure, which is a desirable effect. It is also simpler to calculate; the cosine of a side—and of the respective central angle—is the dot product of the vectors from the sphere center to the endpoints of the side. In a sum over all four sides of all faces, each edge of the graph appears twice. We define the objective function as half that sum, which is the same as the sum of the cosines of the lengths of all edges.

The variables in our optimization are the positions on the unit sphere to which the vertices are mapped. Therefore, *starting values* in this context means an initial mapping of the object's surface to the sphere. It is important for the optimization algorithm that the sphere be com-

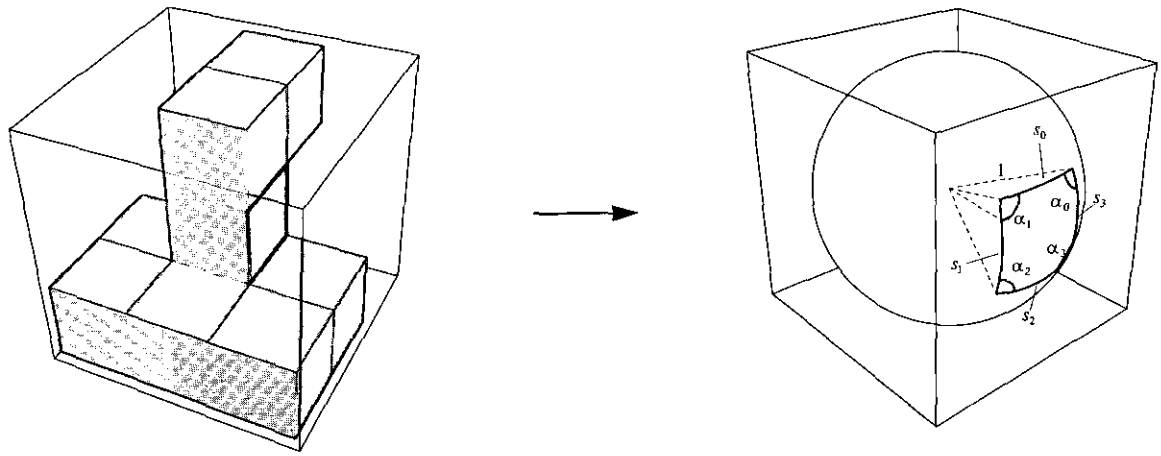

FIG. 2. Every single face on the object's surface is mapped to a spherical quadrilateral. The sides of a spherical polygon are geodesic arcs on the sphere surface. As the sphere has unity radius, the length of a side s_i is equal to the corresponding center angle (in radians). The quadrilateral in this illustration is special in that its four sides $s_0 \cdots s_3$ are equal and its four angles $\alpha_0 \cdots \alpha_3$ are equal: it is the spherical analogue of a square.

pletely covered with faces and none of them overlap. The following subsection describes the construction of an initial parametrization satisfying this requirement. An earlier version of the procedure is described in [11].

3.1. *Initial Parametrization*

The first mapping or parametrization is done in polar coordinates. The two polar coordinates θ and ϕ are determined for all vertices in two separate steps. Two vertices have to be selected as the poles for this process. The choice of these poles is not critical, as the optimization process removes all its influences except a rotation in parameter space. Selecting two poles which lie close together, however, results in a poor initial parametrization. The optimization will converge to the same solution, but it takes more iteration steps. We select the two vertices with maximal and minimal z coordinate in object space, respectively; the y and x coordinates are used to break ties. In our ordering of the surface vertices, they are the first and the last vertex.

A fully worked example with the "two-voxel" object of Fig. 1 is presented in Appendix A. It serves as a concrete and detailed illustration of the general concepts exposed below.

Latitude from Diffusion. Latitude θ should grow smoothly from 0 at the north pole to π at the south pole. In this context, θ is not a free variable but rather an unknown function (of the location on the object) that we are looking for. To assign a latitude value with the desired property to every node, we formulate the corresponding continuous problem as Laplace's equation $\nabla^2\theta = 0$ (except at the poles), with Dirichlet conditions $\theta_{\text{north}} = 0$, $\theta_{\text{south}} = \pi$ for latitude θ [12]. A physical analogy is heat conduction: we heat the south pole up to temperature π, cool the north pole to temperature 0, and ask for the stationary temperature distribution on the heat-conducting surface. As usual in the discrete case, the Laplacian is approximated by finite second differences of the available direct neighbors, which in our case implies that every node's latitude (except the poles') must equal the average of its neighbors' latitudes. These conditions form a sparse set of linear equations, which can be written in the form $A'\boldsymbol{\theta}' = \mathbf{b}'$, where A' is an $n_{\text{vert}} \times n_{\text{vert}}$ matrix, $\theta' := (\theta_0, \theta_1, \ldots, \theta_{n_{\text{vert}}-1})^{\text{T}}$ and $\mathbf{b}'$ is an n_{vert} vector of constants. The border conditions supply two equations and the average property defines $n := n_{\text{vert}} - 2$ equations. Applying the border conditions $\theta_0 = \theta_{\text{north}}$ and $\theta_{n_{\text{vert}}-1} = \theta_{\text{south}}$ results in the reduced $n \times n$ system $A\boldsymbol{\theta} = \mathbf{b}$, where $A = (a_{1,1}, a_{1,2}, \ldots, a_{n,n})$ is symmetric and $\boldsymbol{\theta} := (\theta_1, \ldots, \theta_n)^{\text{T}}$.

The algorithms used to set up the matrix A and the right-hand side vector $\mathbf{b}$ make use of the well-organized surface data structure, as illustrated below in pseudocode notation. The sparsity of the matrix is exploited; only the few nonzero entries (four to seven per row) are stored. This saving is essential for the treatability of larger objects.

```
Set up matrix A:
  for vertex = 1 . . . n
    a_vertex,vertex := number of direct neighbors;
    for the direct neighbors of vertex
      if the neighbor is not a pole
        a_vertex,neighbor := −1;

Set up constant vector b:
  set all entries of b to 0;
  for the direct neighbors of south_pole
    b_neighbor := π;
```

This sparse symmetric linear system of equations is solved with the PILS package [13]. The solution has the important property that latitude θ varies monotonically between the poles, since there can be no local extremum by virtue of the maximum principle [14]. Figure 3a shows the resulting θ for a simple test object, "duck."

Longitude from Diffusion. Unlike latitude, longitude is a cyclic parameter: When we walk around a sphere counterclockwise (seen from the north), longitude keeps increasing monotonically all the time, but there must be a place where longitude leaps back by 2π. A global longitude parameter always has a discontinuous line running from pole to pole, and the step height is 2π. Consider, as an analogy, local time on every spot of the globe; the date line is a 24-h discontinuity, but not a meridian. In our problem, the choice of the date line is immaterial: it just has to connect the two poles. The data line is chosen as a path with steepest latitude ascent in each of its nodes. The values crossing the date line from west to east are decremented by 2π, values propagated to the west are incremented by 2π. The poles and all links to them are removed from the net, making the topology of the net that of a tube. Longitude remains undefined for the poles. The cyclic Laplace equation $\nabla^2\phi = 0$ (with date line) again corresponds to a system of linear equations in the discrete case.

The new system of linear equations is structurally identical to that for latitude. Typically, only a small part of the equations is different. The matrix for longitude ϕ differs from that for latitude θ only by the values of six diagonal entries, corresponding to the three neighbors of each pole. This similarity simplifies setting up and solving the new system.

Due to the cyclic boundary conditions, the solution ϕ is defined only up to an additive constant. The linear equations are dependent, and the system is singular. To make it regular, we have to specify the longitude of one vertex. We arbitrarily set $2\phi_1 = 0$. This equation can be added to any row of the system. We add it to the first row.

The following portions of pseudo-code update the matrix and generate a new right-hand side vector.

```
Modify matrix A:
  for both poles
    for the direct neighbors of pole                    cut link to pole
      a_neighbor,neighbor -= 1;
  a_0,0 += 2;                                           for φ_1 = 0

Set up constant vector b:
  for row := 1 . . . n do
    b_row := 0;
  previous := north pole;
  here := 1;                                            any neighbor of north pole
  maximum := 0.0;
  while (here != south pole)                            walk on date line
    for the direct neighbors of here
      if θ_neighbor > maximum then
        maximum := θ_neighbor;
        nextpos := position of neighbor;
      if neighbor == previous then
        prevpos := position of neighbor;
    for the direct neighbors clockwise between prevpos and nextpos do
      add 2π to b_neighbor;
      subtract 2π from b_here;                          totally: 2π · # west_neighbors
    previous := here;
    here := neighbor of here indicated by nextpos;
```

In spherical coordinates, longitude is undefined at the poles. We arbitrarily set $\phi_{north} = \phi_{south} = 0$. Figure 3b illustrates the resulting ϕ for the duck test object.

For every node of the net, we now have computed a latitude θ and a longitude ϕ. This defines a continuous, unique mapping from the surface of the original object to the surface of a sphere. The spherical parameters ϕ and θ can also be used in a flat Cartesian coordinate system as shown in Fig. 4b, which gives an overview of the whole unfolded net. The right border wraps around to the left

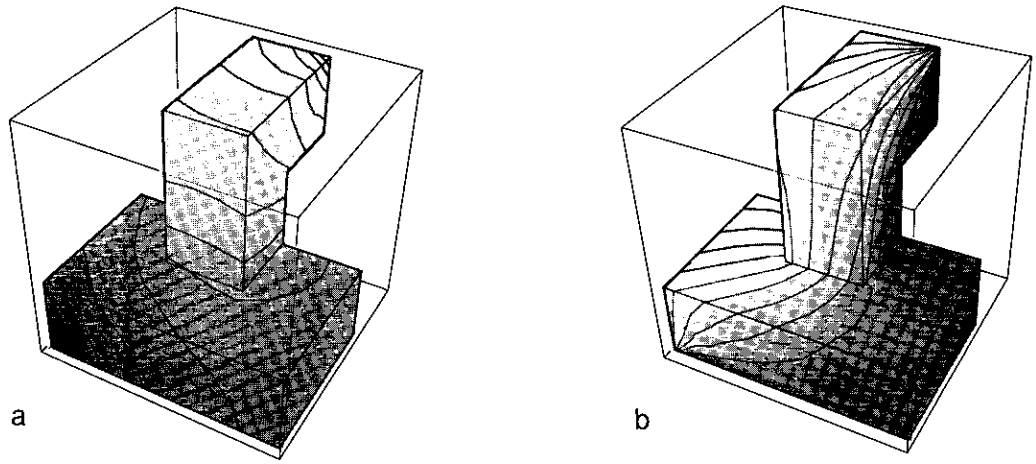

FIG. 3. The simple object "duck" consisting of nine voxels is used for illustrating the initial parametrization. The north pole is at the lower left, the south pole at the upper right. Latitude is mapped on the object's surface as a gray value in (*a*); isolatitude lines are drawn every $\pi/16$. Longitude is shown in (*b*); isolongitude lines ("meridians") are $\pi/8$ apart.

border, and vice versa. The top and bottom borders stand for the poles. The θ axis points down to produce the same orientation as in the other figures; the south pole is at the top. The pole coordinates are transformed to Cartesian and yield starting values for the optimization.

3.2. *Optimization Method*

Powerful methods for nonlinear constrained minimization exist [15]. The commonly available optimization routines cannot be used for larger objects because they are not suited for such a large problem, as they do not exploit its sparsity and information available about the constraints. A Newton–Lagrange algorithm [15] based on the sparse linear solver package PILS [13] is used to find the constrained maximum of the goal function. An active set strategy enforces the inequality constraints. A detailed discussion of the optimization algorithm would exceed the scope of this paper. In fact, any other constrained optimizer that could handle the size of the problem could be substituted.

The question arises if a solution exists at all and if it is unique. The goal function is a sum of cosines, which cannot exceed 1. Thus the goal function is bounded by the number of edges in the graph, $2n_{\text{faces}}$. Any constraints can only lower the attainable maximum, but never make

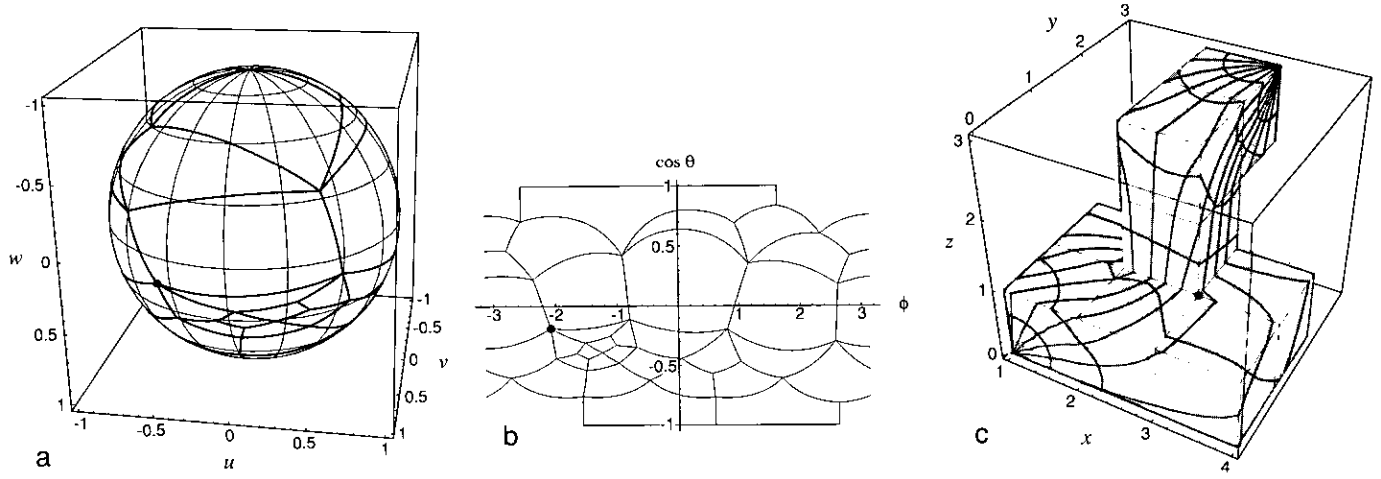

FIG. 4. Diffusion yields the initial parametrization, which is visualized in three different ways. (a) The surface net is plotted on the spherical parameter space. The thick lines depict the edges of the original square faces. The equidistance for both θ and ϕ is $\pi/8$. (b) ϕ and cos θ are interpreted as Cartesian coordinates. The monotonic cosine function is applied to give a true-area cylindrical projection. The horizontal lines at ± 1 are the poles. (c) Conversely, the globe coordinate grid is drawn over the object. For comparison, one vertex is marked with a black dot in all diagrams.

the goal function unbounded. Another problem could arise from incompatible constraints. From their definition—vector length, area content, angle—they are not inherently incompatible. There are $3n_{\text{vert}}$ degrees of freedom but only $n_{\text{vert}} + n_{\text{face}} - 1 = 2n_{\text{vert}} - 3$ equations to satisfy. At the solution, only few (typically less than five) inequalities are active. We have no indications that there might be any local maxima. Of course any rotated version of the solution yields the same values for the goal function and for all constraints, but this is not considered an essentially different solution.

Constraining all angles to the range $[0, \pi]$ makes sure none of the quadrilaterals can fold over. Thus the neighborhood relations are preserved during the optimization process.

As the goal function, the constraints, and their derivatives have to be calculated repeatedly, it is important that we can use an efficient data structure that holds information about adjacency (for the goal function and matrix sparsity) and surface facets (for the area constraints).

The solution of the nonlinear program defines the optimal parameterization of our object's surface. Figure 4 shows the starting point for the optimization; the right diagram is a combination of those in Fig. 3. Figure 5 visualizes the result of the optimization in different ways. The same vertex as in Fig. 4 is marked.

Figure 6 shows several stages of the optimization process. The starting configuration, "0," corresponds to Fig. 4a. The final result is obtained after 64 iterations; it has the label "64" and is the same as Fig. 5a.

4. SPHERICAL HARMONIC SHAPE DESCRIPTORS

Let x, y, and z denote Cartesian object space coordinates and θ and ϕ polar parameter space coordinates. The parameterization gives us three explicit functions defining the object surface as

$$\mathbf{x}(\theta, \phi) = \begin{pmatrix} x(\theta, \phi) \\ y(\theta, \phi) \\ z(\theta, \phi) \end{pmatrix}.$$

When the free variables θ and ϕ run over the whole sphere (e.g., $\theta = 0, \ldots, \pi$; $\phi = 0, \ldots, 2\pi$), $\mathbf{x}(\theta, \phi)$ runs over the whole surface of our object. As for our shape description method, we use the surface net representation to expand a 3-D shape into a complete set of basis functions. We use the spherical harmonics; Y_l^m denotes the function of degree l and order m [16]. Definitions are given in the Appendix. The coefficients

$$\mathbf{c}_l^m = \begin{pmatrix} c_{x_l^m} \\ c_{y_l^m} \\ c_{z_l^m} \end{pmatrix}$$

in this series are three-dimensional vectors. Their components, $c_{x_l^m}$, $c_{y_l^m}$, and $c_{z_l^m}$, are usually complex numbers; they are real numbers for $m = 0$. The series takes the form

$$\mathbf{x}(\theta, \phi) = \sum_{l=0}^{\infty} \sum_{m=-l}^{l} \mathbf{c}_l^m \, Y_l^m(\theta, \phi). \tag{1}$$

The coefficients of the spherical harmonic functions of different degrees provide a measure of the spatial fre-

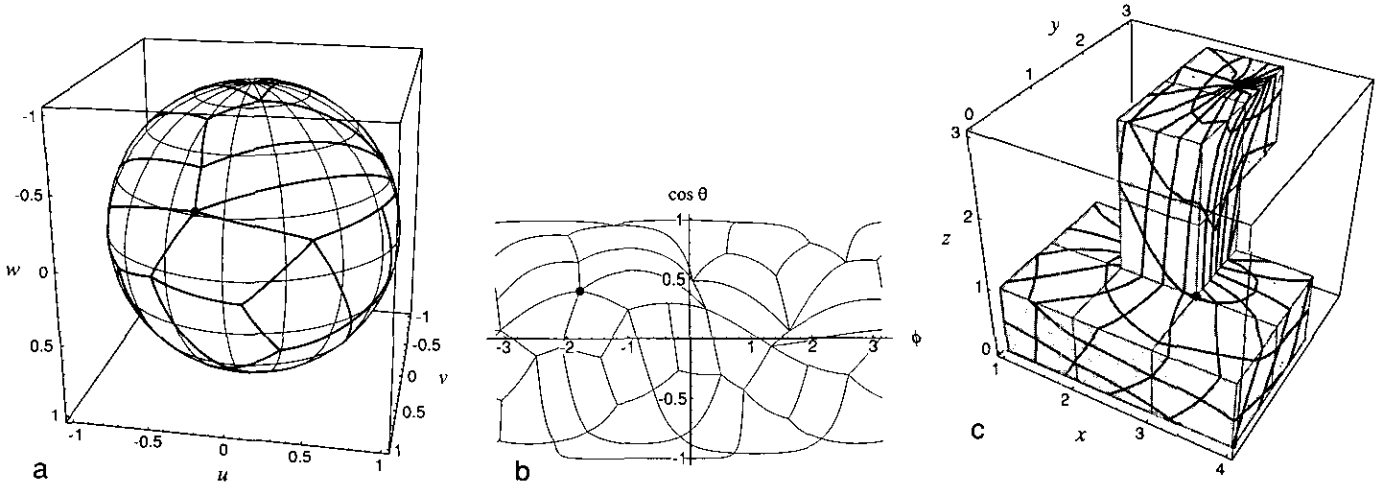

FIG. 5. The result of the optimization, plotted in the same ways as in the previous figure. The areas of elementary facets in parameter space are now equal, and local distortions are minimized. The rotational position of the net on the sphere is arbitrary. (c) After the optimization, the former poles have lost their prominent role. They have now the same importance as any other point in parameter space and could lie anywhere on the surface. Only the use of polar coordinates for visualization gives them a conspicuous appearance.

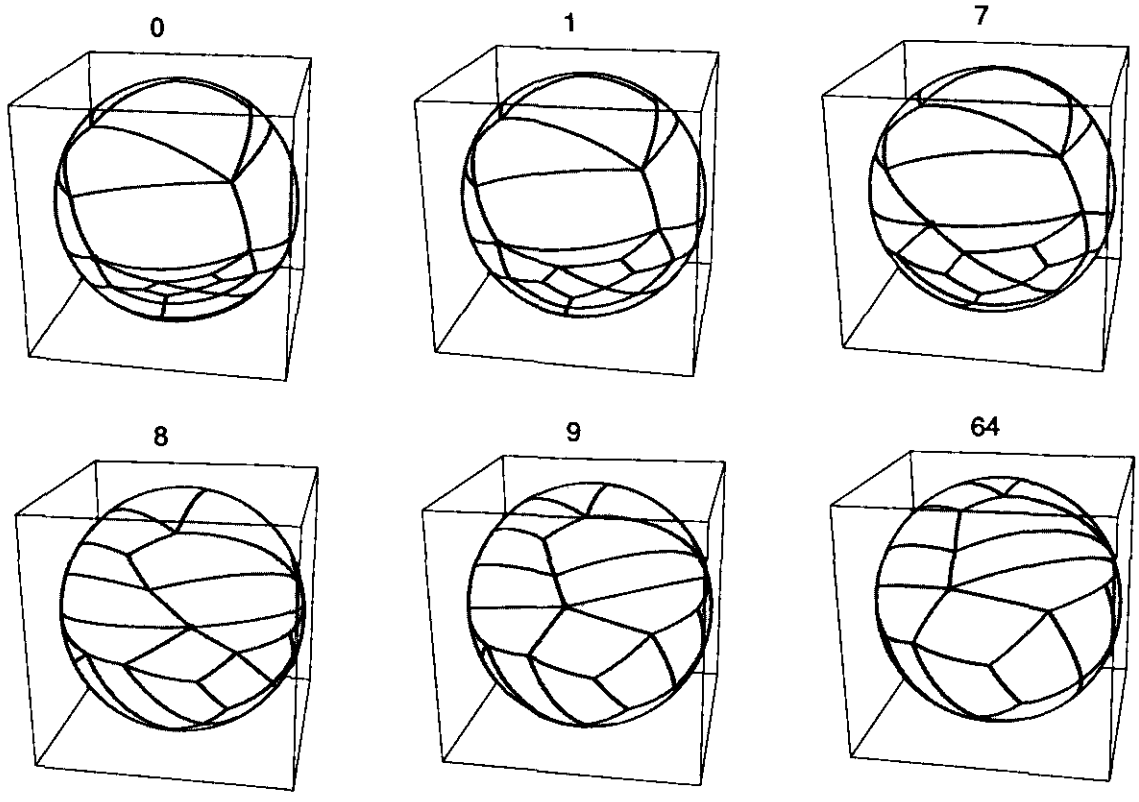

FIG. 6. Stages from an optimization run. The labels indicate the number of iterations. "0" marks the starting configuration, where the areas of the facets vary considerably. "64" is the final result; all facets have now equal area, and distortions are minimized.

quency constituents that compose the structure. Partial sums of the series (1) are plotted in Fig. 7. The sums are truncated by limiting l to $0 \leq l < n_l$, where $n_l = 2, 4, 8$. As higher frequency components are included, more detailed features of the object appear.

The use of orthogonal basis functions is convenient for the calculation of the expansion coefficients. Formally, the coefficients are calculated by forming the inner product of $\mathbf{x}$ with the basis function in question:

$$\begin{aligned} \mathbf{c}_l^m &= \langle \mathbf{x}(\theta, \phi),\, Y_l^m(\theta, \phi) \rangle \\ &= \int_0^{\pi} \int_0^{2\pi} \mathbf{x}(\theta, \phi)\, Y_l^m(\theta, \phi)\; d\phi \sin\theta\; d\theta. \end{aligned} \tag{2}$$

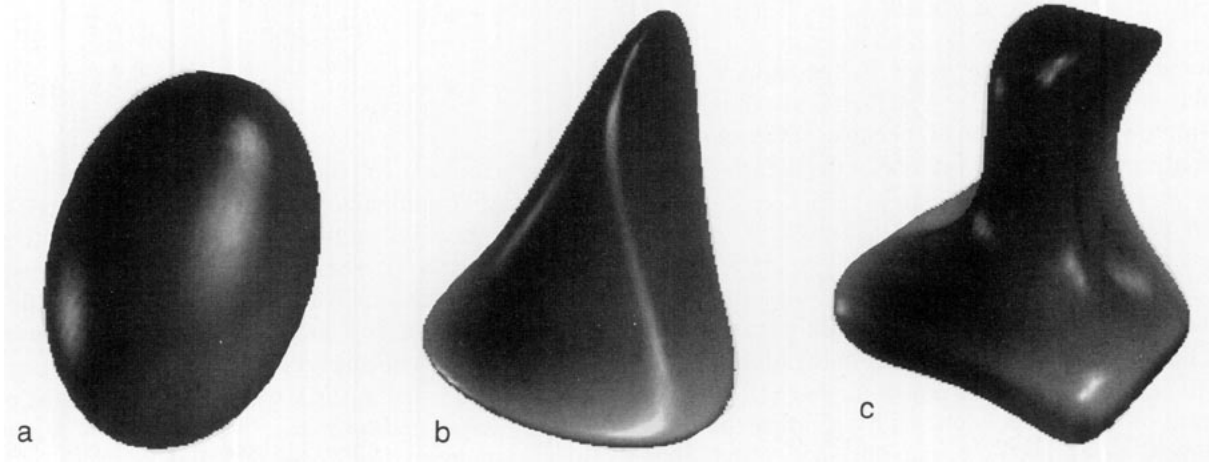

FIG. 7. Global shape description by expansion into spherical harmonics: The figures illustrate the reconstruction of the partial spherical harmonic series, using coefficients up to degree 1 (a), to degree 3 (b), and to degree 7 (c).

Note that the parametrization defines the function $\mathbf{x}(\theta, \phi)$ only for the parameter coordinates of the vertices. Let n_{vert} denote the number of vertices and i be the index of a vertex, $0 \leq i < n_{\text{vert}}$. Only $\mathbf{x}(\theta_i, \phi_i) = \mathbf{x}_i$ is defined. For the evaluation of the integral (2) we would have to define an interpolating function between these sample points; an adaptation of bilinear interpolation could be used for this purpose. But this would introduce an artificial subvoxel resolution that has no base in the input data. On the other hand, the straightforward discretization of the integral,

$$\mathbf{c}_l^m \approx \frac{4\pi}{n_{\text{vert}}} \sum_{i=0}^{n_{\text{vert}}-1} \mathbf{x}_i Y_l^m(\theta_i, \phi_i), \tag{3}$$

does not give the precise coefficients of a series representing our object. The reason is that although the functions Y_l^m are orthonormal, their values evaluated at some set of parameter pairs (θ_i, ϕ_i) will generally not form an orthonormal set of vectors. We adopt some indexing scheme $j(l, m)$ that assigns a unique index j to every pair l, m, like, e.g., $j(l, m) := l^2 + l + m$. When the degree of the spherical harmonics is limited, i.e., $0 \leq l < n_l$, j is also limited by $j < n_j = n_l^2$. We can arrange all needed values of our basis functions in an $n_{\text{vert}} \times n_j$ matrix B, where $b_{i,j(l,m)} = Y_l^m(\theta_i, \phi_i)$. In the usual case where n_j is significantly smaller than n_{vert}, the columns of B are approximately orthogonal. We further arrange the object space coordinates of all vertices in an $n_{\text{vert}} \times 3$ matrix $X = (\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_{n_{\text{vert}}-1})^{\mathrm{T}}$ and all coefficients in the $n_j \times 3$ matrix $C = (\mathbf{c}_0^0, \mathbf{c}_1^{-1}, \mathbf{c}_1^0, \ldots)^{\mathrm{T}}$. The equations (3) for all l and m take the compact form $C \approx (4\pi/n_{\text{vert}})B^{\mathrm{T}}X$. But what we really want is a spherical harmonic series that passes near the real positions of our vertices, i.e., $X = BC + E$, where the error matrix E should be small. These so-called normal equations are solved with least-square sums over the columns of E by

$$C = (B^{\mathrm{T}}B)^{-1}B^{\mathrm{T}}X. \tag{4}$$

The global approximation error is the square of the Frobenius norm of $E = BC - X$, which is also minimized. This is not too different from (3) because of the symmetric $n_j \times n_j$ matrix $(4\pi/n_{\text{vert}})B^{\mathrm{T}}B$ is close to the identity matrix.

4.1. *Rotation Independent Descriptors*

The coefficients obtained thus far still depend on the relative position of the parameter net of the object surface and on the orientation of the object in space (Fig. 5). We can get rid of these dependencies by rotating the object to canonical positions in parameter space and object space. This needs three rotations in parameter space and three rotations in object space, when rotations are described by Euler angles. All rotations result in new linear combinations of the components of the harmonic descriptors.

The relations between the Cartesian and the spherical coordinates of the parameter space are $u = \sin\theta \cos\phi$, $v = \sin\theta \sin\phi$, and $w = \cos\theta$. To define a standard position we consider only the contribution of the spherical harmonics of degree $l = 1$ in Eq. (1), an ellipsoid (see Fig. 7):

$$\mathbf{x}_1(\theta, \phi) = \sum_{m=-1}^{1} \mathbf{c}_1^m Y_1^m(\theta, \phi). \tag{5}$$

This sum involves the basis functions $Y_1^{-1} = \sqrt{3/8\pi}(u - iv)$, $Y_1^0 = \sqrt{3/4\pi}\,w$, and $Y_1^1 = \sqrt{3/8\pi}(u + iv)$. Any three real valued linear combinations of these (i.e., $(c_{x_1^{-1}})^* = -c_{x_1^1}$ and $c_{x_1^0} \in \mathbb{R}$ and similarly for y and z), interpreted as Cartesian coordinates in the object space, will always describe an ellipsoid. We rotate the parameter space so that the north pole ($\theta = 0$, on the w axis) will be at one end of the shortest main axis of this first-order ellipsoid and the point where the Greenwich meridian ($\phi = 0$) crosses the equator ($\theta = \pi/2$, on the u axis) is at one end of the longest main axis.

At the three main axes, the length of the vector $\mathbf{x}_1(\theta, \phi)$ is stationary: it reaches a maximum, a saddle point, and a minimum, respectively. Measuring Euclidean lengths becomes simpler when we transform the component vectors to a Euclidean, real valued form. A short calculation yields

$$\mathbf{x}_1(\mathbf{u}) = A\mathbf{u} = A\begin{pmatrix} u \\ v \\ w \end{pmatrix} = \mathbf{a}_1 u + \mathbf{a}_2 v + \mathbf{a}_3 w, \tag{6}$$

where

$$\begin{aligned} A &= (\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3) \\ &= \frac{\sqrt{3}}{2\sqrt{2\pi}}(\mathbf{c}_1^{-1} - \mathbf{c}_1^1, i(\mathbf{c}_1^{-1} + \mathbf{c}_1^1), \sqrt{2}c_1^0). \end{aligned} \tag{7}$$

We are looking for the unit vectors $\hat{\mathbf{u}}_1$, $\hat{\mathbf{u}}_2$, and $\hat{\mathbf{u}}_3$ that maximize or minimize the length of the vector. The solutions are the eigenvectors of $A^{\mathrm{T}}A$, with nonnegative eigenvalues $l_1^2 > l_2^2 > l_3^2$. Their roots l_1, l_2, and l_3 represent half the lengths of the main axes of the ellipsoid. At the middle eigenvector, $\hat{\mathbf{u}}_2$, $\|\mathbf{x}_1\|$ has a saddle point rather than an extremum. The normalized eigenvectors form the rotation matrix $R_{uvw}^{\mathrm{T}} = (\hat{\mathbf{u}}_1, \hat{\mathbf{u}}_2, \hat{\mathbf{u}}_3)$, which is applied to the parameters $\mathbf{u}_i$ associated with each vertex i: $\mathbf{u}_i' = R_{uvw}^{\mathrm{T}}\mathbf{u}_i$. This new parametrization results in new coefficients $\mathbf{c}_l^{m\prime}$ and hence in the new coefficient matrix $A' = AR_{uvw}$. Its three column vectors $\mathbf{a}_1'$, $\mathbf{a}_2'$, and $\mathbf{a}_3'$ are the main axes of the first-order ellipsoid in object space.

All rotations are determined based on the values of $\mathbf{c}_1^i$ ($i = -1, 0, 1$) of the ellipsoid only, but they are applied to all components of the descriptor $\{\mathbf{c}_l^m\}$. The parameter space rotations result in a different description of the same object in the same position, just parametrized in a standard way.

Now, the ellipsoid is rotated in the object space to make its main axes coincide with the coordinate axes, putting the longest ellipsoid axis along x and the shortest one along z. The object space rotations require only the matrix multiplication $\mathbf{c}_l^{m\prime\prime} = R_{xyz}\mathbf{c}_l^{m\prime}$. The object space rotation matrix is $R_{xyz} = \text{diagonal}\,(1/l_1, 1/l_2, 1/l_3) \cdot A'^T$. It rotates the main axes of the ellipsoid into an axis-parallel position and makes the coefficient matrix $A'' = R_{xyz}A' = R_{xyz}AR_{uvw}$ diagonal. The elements of the diagonal are the lengths of the main axes of the ellipsoid.

The descriptors $\mathbf{c}''$ are now invariant under rotation of the object, except mirrorings (rotations by π). Including information from higher degree coefficients could eventually disambiguate these cases. Ignoring $\mathbf{c}_0^0$ results in translation invariance. Scaling invariance can be achieved by dividing all descriptors by l_1, the length of the longest main axis.

4.2. *Importance of Uniform Parametrization*

The report thus far presumed that a homogeneous density and a minimal distortion of the parameter net would be important for shape characterization, especially for obtaining an invariant description. Similarly, the 2-D expansion of contours $s(t)$ into series of harmonics [5] was based on the model of tracing a curve with constant velocity, i.e., assigning same lengths ΔL to equivalent parameter steps Δt. A nonuniform distribution of parameters on an object surface, e.g., by clustering at certain locations, seems to be suboptimal with respect to a uniform representation of the whole surface. One would expect an overrepresentation of some parts to the disadvantage of others, resulting in distorted shape descriptions.

The importance of a parametrization with minimal distortion can be demonstrated with an experiment. The expansion into a series of spherical harmonics is calculated both for the nonuniform initial parametrization (bypassing the optimization step for this part of the experiment) and for the result after optimization. A manifestly non-star-shaped form was chosen: the original object consists of 11 voxels and is shaped like the character E. Its initial and optimized parametrizations are given in Fig. 8 for comparison. The diagrams correspond to Figs. 4a and 5a.

Figure 9 illustrates the expansion in a spherical harmonic series up to degree 10 and the truncated reconstruction up to degree 1 (top), 4 (middle), and 10 (bottom) for the initial (left) and optimized (right) parametrization. (A fivefold oversampling was applied to the surface to represent it accurately. This may be viewed as a rough form of numerical integration.) Comparing the expressive difference, one can conclude that a uniform parametrization is absolutely essential to obtain useful spherical harmonic descriptors. Even from the distorted initial parametrization, descriptors can be derived that are "optimal" in the least-squares sense, but the series of harmonics does not reflect the shape properties of the surface. Using the optimized parametrization, the reconstructed objects demonstrate the most desirable behavior that coefficients of higher degree add information about details of higher spatial frequency. The first-degree harmonic approximation (Fig. 9b) covers the whole object and comprises information about the major size and elongation, whereas the three "legs" of the E-shape appear in the reconstruction using harmonics up to degree 4 (Figs. 9d and 9f).

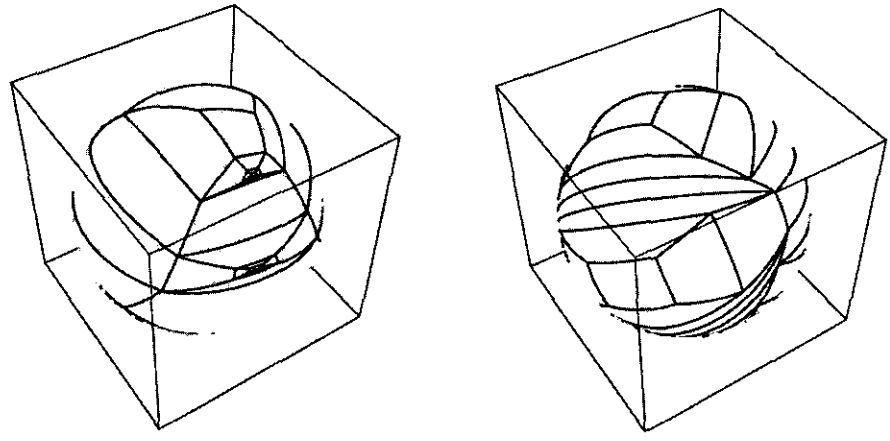

FIG. 8. Two different parametrizations of the "E" object. (Left) The initial parametrization is the starting point of the optimization. (Right) The optimized parametrization.

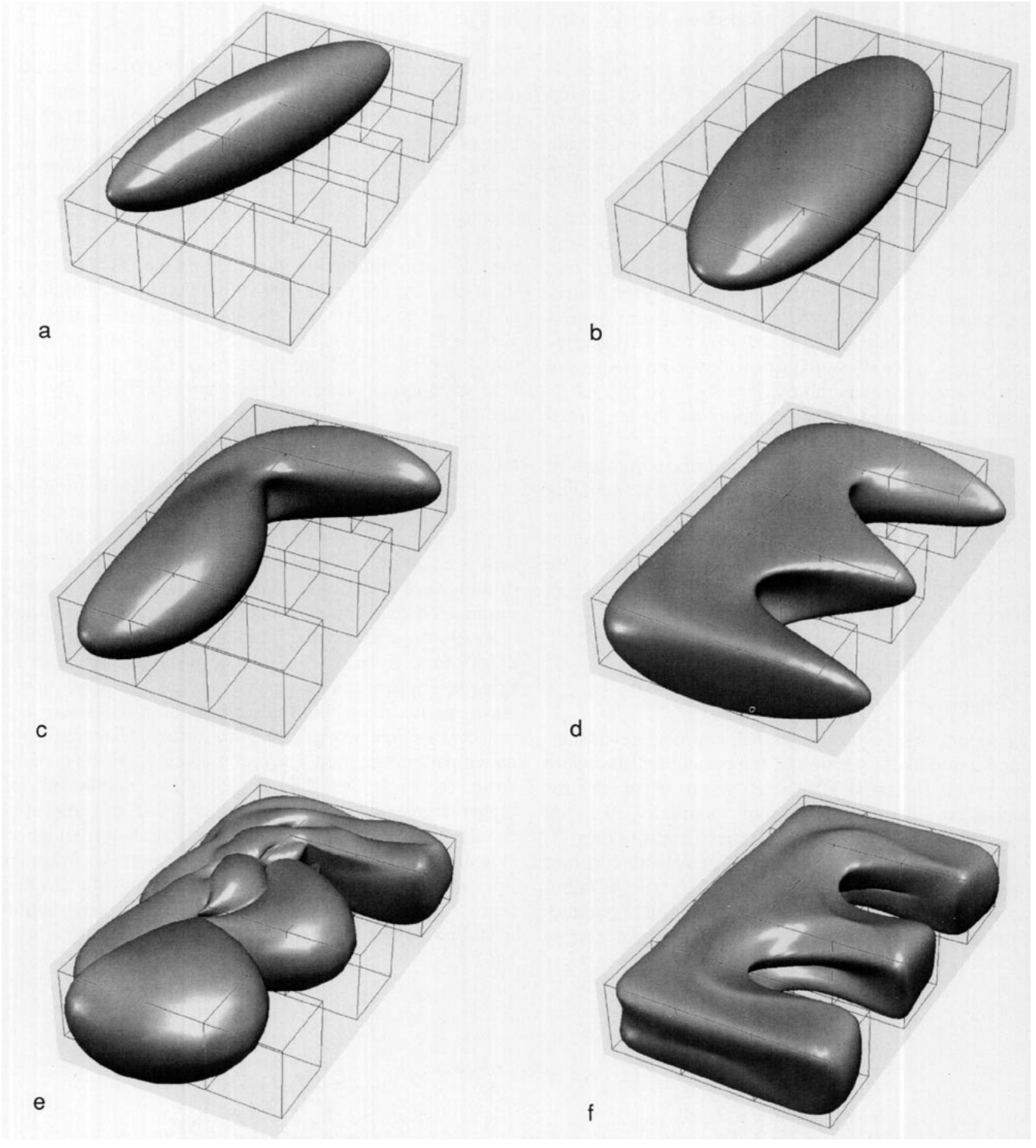

FIG. 9. Experiment demonstrating the importance of a homogeneous parameter distribution for shape description. The original E-shaped object surface, indicated by a wireframe, is expanded into a series of spherical harmonics one starting from a nonuniform parametrization and the second from an optimized parametrization. The series are truncated at degree 1 (a, b), 4 (c, d), and 10 (e, f). Shaded surfaces depict the reconstructions. The images in the left column (a, c, e) clearly illustrate the poor shape representation based on nonuniform parametrization. A significant improvement is achieved by using the optimized parametrization (b, d, f), reflecting the hierarchical nature of a harmonic approximation regarding shape details at different scales. Quantitatively, $\|E\|_F$ measures the error, which is divided by the square root of the number of rows of E to give the rms distance in pixel units. The values for the individual reconstructions are 1.143 (a), 0.884 (b), 0.729 (c), 0.250 (d), 0.313 (e), and 0.102 (f).

FIG. 10. Test objects and results. From Top to Bottom: objects "box A," "box B," "c4," "c8," "patella," and "ventricle." From left to right in each row: The cuberille interpretation of the input data, the parametrization drawn as a flat net spread out in (ϕ, $w = \cos\theta$) parameter space, and the reconstruction superimposed with a wireframe of the original data.

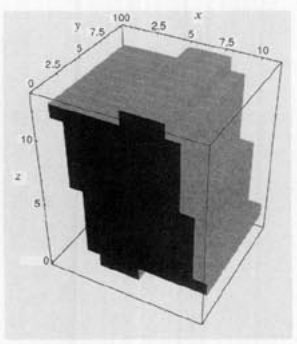
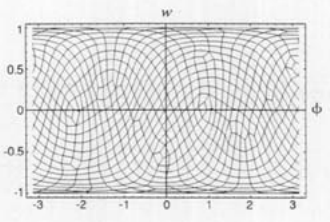
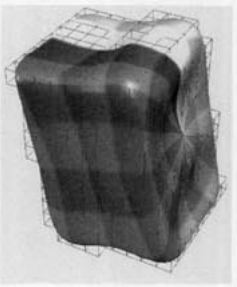
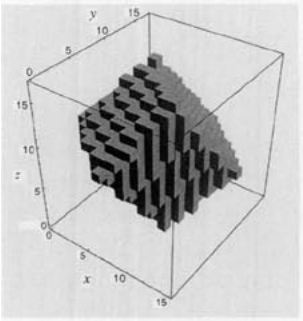
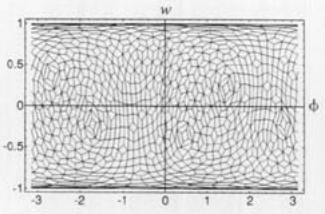
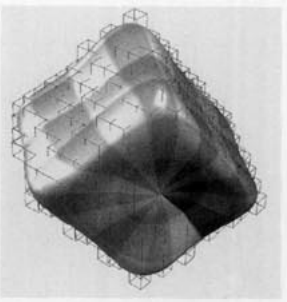
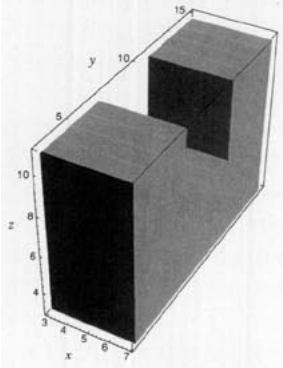
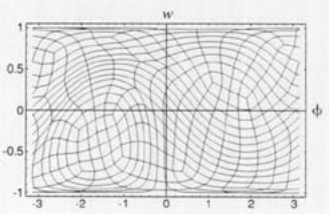
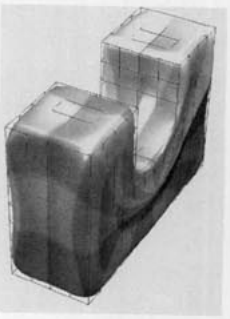
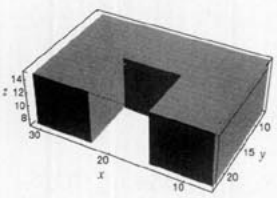
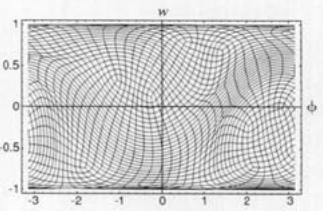
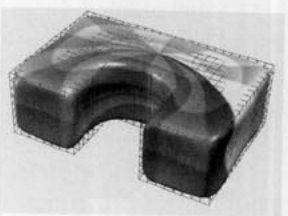
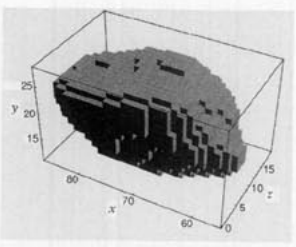
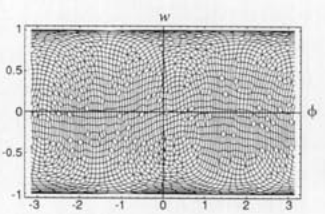
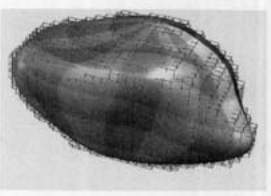
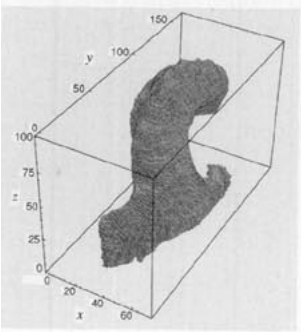
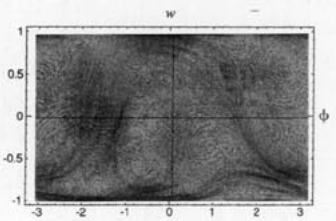


TABLE 1
Comparison of the Six Test Objects, Including the Squared Euclidean Distances between Their Descriptors

Name	n_{vert}	Time	Squared Distance to Box A	Box B	c4	c8	Patella	Ventricle
Box A	628	33 s	0	0.0241	0.2370	0.2378	0.0673	0.3850
Box B	902	267 s	0.0241	0	0.2796	0.2808	0.0859	0.4555
c4	354	26 s	0.2370	0.2796	0	0.0002	0.2309	0.2175
c8	1410	338 s	0.2378	0.2808	0.0002	0	0.2299	0.2143
Patella	2182	536 s	0.0673	0.0859	0.2309	0.2299	0	0.2623
Ventricle	37654	28 h	0.3850	0.4555	0.2175	0.2143	0.2623	0

4.3. *Experimental Results*

The surfaces of the following test objects are parametrized, and their shapes are expressed through spherical harmonic descriptors.

c4 A "c"-shaped polyhedron made up from five $4 \times 4 \times 4$ voxel cubes.

c8 The same object, magnified by a factor of 2 in all coordinate directions.

Box A The voxels fill a rectangular box, which is not aligned with the coordinate axes.

Box B Rotating box A results in a completely different sampling.

Patella The patella was extracted from a segmented CT scan data set of a human knee.

Ventricle The ventricular system has been segmented from a MRI data set of a hydrocephalus patient.[1] We selected one lateral ventricle. The data are interpolated to compensate for the aspect ratio of 1 : 1 : 6.4 of the original data.

For each object, Fig. 10 presents the cuberille interpretation of the input data, a spread-out graph of the parametrization, and the reconstruction from spherical harmonic descriptors. The cylindrical projection we chose for drawing the parametrization shows the true area ratios. The smooth surface of the reconstruction is shaded in a pattern that makes possible the estimation of the parameter values. These latter parameter values do not coincide with those in the middle diagram. They rather differ by the rotation in parameter space that makes the descriptors rotation invariant. The object space rotation is suppressed in the diagrams to show the spatial relation of the original data—shown as a wireframe—with the reconstruction from the descriptor, up to degree 8. In the case of the ventricle, this shows an insufficient degree of detail, but the same value was chosen for comparability.

Table 1 summarizes the sizes and differences of the various test objects. Virtually all of the processing time for an object is spent in the optimization. The figure for computation time must be interpreted with caution; it qualifies only the optimization program, which is not necessarily as efficient as possible, and which might be substituted with an out-of-the box optimizer. Times are measured on a HP 9000/735. The number of vertices, n_{vert}, indicates the size of the problem: the optimization has $3n_{vert}$ variables, $2n_{vert} - 3$ equality constraints, and $4n_{vert} - 8$ inequality constraints.

The distance between the descriptors appears to be a valid rough measure of shape dissimilarity. The matrix of distances is symmetric by definition. The two c's are most similar to each other. The two boxes are also quite similar. Both these examples illustrate the translation, rotation, and scale invariance of the descriptors. The patella is more similar to a box than to a c, whereas the ventricle is more similar to a c than to any of the other objects.

5. CONCLUSIONS

This paper presents new techniques to generate explicit parametric representations of *convoluted* object surfaces with minimal distortion and to characterize 3-D surfaces by *invariant* spherical harmonic shape descriptors. We intended to overcome traditional limitations of expressing an object surface by explicit parametric representations. These are restricted to star-shaped objects, have nonuniform spacing of parameters on the object surface, and often require a specific choice of the new coordinate system with respect to the object geometry. Our parametrization technique is valid for closed surfaces of simply connected objects, but it generalizes naturally to the unfolding or flattening of complex surface patches (open surfaces with one edge). Other classes of simple 3-D surfaces with

[1] Data courtesy of Ron Kikinis, Surgical Planning Lab, Department of Radiology, Brigham and Women's hospital and Harvard Medical School, Boston.

different topology, for example tori and tubes, are not considered.

The following paragraphs summarize the properties of the novel surface analysis techniques.

Parametrization. The unfolding and optimization procedure maps nodes onto a sphere, each of which can be expressed by two parameters and is associated with a voxel vertex. The procedure imposes no restrictions regarding the geometry of objects and is suitable for surfaces of arbitrary complexity. An initial diffusion of "temperature" on the object's surface achieves a continuous mapping by assigning latitude and longitude to each surface vertex. The position of poles and the geometry of the object produce a clustering of parameters at certain regions of the object surface. This nonuniform distribution of parameter density on the object surface is corrected by a nonlinear optimization technique, which equalizes the area of original surface elements in parameter space and minimizes their distortion. The latter is formulated as the goal function of the nonlinear optimization problem and the former as its constraints. The resulting arrangement of the vertex nodes on the sphere (parameter space) reflects the geometry of the original shape and achieves similar parametrizations for similar shapes, but is free to rotate around any axis as no surface points are kept fixed. It must be pointed out that the polar coordinate system for spherical surfaces is only used for the sake of visualization. This coordinate system itself determines a nonuniform tessellation of the sphere and may give a misleading visual impression of parameter densities (see, e.g., Fig. 5). Alternative display techniques could be found by tessellating the sphere into small cells, for example by regular or semiregular polyhedra.

The parametrization of object surfaces forms an intermediate representation with the following properties:

- The surface of arbitrarily shaped (but simply connected) objects can be parametrized. Objects are not restricted to a limited family of shapes; even protrusions and intrusions are appropriately dealt with.
- The surface is explicitly represented by the variation of two parameters, expressing the properties of local surface neighborhoods as well as of the global shape.
- The parametrization results in a continuous (no overlap of elements), one-to-one mapping of surface vertices to a sphere. While varying the two parameters θ and ϕ over the parameter range, each point of the surface is visited exactly once.
- The optimization results in a unique, reproducible solution (except for rotation).
- The parametrization preserves area exactly and minimizes local distortions which cannot be avoided when mapping an object with corners to a sphere (see location marked by a black dot in Figs. 5a–5c). The uniformity of the parametrization is important for a subsequent shape description, as illustrated in Fig. 9.

The parametrization technique is potentially interesting for applications where a mapping of convoluted object surfaces to a simple surface—like the sphere—is required. The unfolding, or flattening, process with minimization of distortions generates a representation which could serve as a useful intermediate surface description for many structure analysis processes. The only restriction, i.e., the presence of the closed surface of a simply connected object, highlights the generality of the approach.

As discussed previously, unfolding is in principle not restricted to closed surfaces. Ongoing developments focus on a similar technique for flattening parts of surfaces onto planar charts. This procedure could be interesting for the comparative analysis and description of convoluted surface patches. Practical applications can be found in brain research, for example, where regional cortical patterns of the human brain are qualitatively and quantitatively analyzed.

Applications were first constrained by the efficiency of the nonlinear optimization. Although the method itself poses no restriction on the maximum number of object vertices, the commonly available optimization routines cannot be applied for a large number of vertices (exceeding several hundreds). In real applications, e.g., the analysis of volume data in medicine, one can expect to deal with object surfaces with up to one-million voxel vertices. We have recently developed an optimization technique which takes into account the sparsity of the problem and the specific nature of the local constraints.

Shape Description. The new parametrization makes possible representation of object surfaces of arbitary complexity. As one possible approach to global shape analysis, it enables us to expand an object surface into a series of spherical harmonic functions. The numerical coefficients in the Fourier series represent an object-centered, surface-oriented descriptor of the object's form. Surface description with harmonic descriptors is no longer restricted to star-shaped objects but can now be applied to a broad class of shapes. Invariance has to be considered as one of the most important properties of shape description, as it only allows a comparative analysis between different objects or a match between objects and models. With the development of new scale and rotation independent descriptors we obtain a *global, object-centered shape description* which is invariant to standard transformations (rotation, translation, and scaling). The invariant postitioning of the object and of the parameter net are based on the analysis of harmonic descriptors up to the first degree, defining the three main axes of the ellipsoid. The symmetry of this low-frequency representation deter-

mines a general 3-D object only up to four different positions. Including coefficients of higher degree, for example at the extremal points of the description up to degree 1, could disambiguate the different cases and avoid a matching using four different object descriptions.

Applications of global object representation and description in computer vision and image analysis are imminent. Generality with respect to object complexity, invariance to standard transformations, and descriptive power in terms of object geometry are the critical issues for shape-based categorization and comparison of 3-D objects. Robot vision and medical image analysis, e.g., are dealing with recovering the global shape characteristics of objects. Whereas the former most often deals with a small number of views of objects and hence only a partial surface description, modern scanning techniques in medicine can provide full 3-D images. Mapping of convoluted surface structures and high-level 3-D shape descriptions of anatomical objects (e.g., the heart cavities, the ventricular system, or cortical substructures of the brain) will play a significant role in the analysis of shape dissimilarities and morphological deformations and in the comparison of malformed with "normal" shape structures. The overall shape is captured by a small number of parameters, expressing structural details at various scales with coefficients of different degrees. The continuous analytical description of the approximated surface makes it possible to compute local differential characteristics, e.g., principal curvature [17]. Inferring the differential structure would result in a characterization of important landmarks, e.g., for registration of different 3-D objects.

APPENDIX A: AN EXAMPLE OF THE INITIAL PARAMETRIZATION

The "two-voxel" object (cf. Fig. 1) is used throughout this example. The surface data structure of this object and a flat diagram of the surface net (Fig. A1) are given below for reference.

Node number	x	y	z	Neighbors
0	{{{ 9,	14,	6},	{1,7,6,9,3,4}},
1	{{ 10,	14,	6},	{0,3,4,5,2,8,7,6}},
2	{{ 11,	14,	6},	{1,4,5,11,8,7}},
3	{{ 9,	15,	6},	{4,1,0,6,9,10}},
4	{{ 10,	15,	6},	{3,9,10,11,5,2,1,0}},
5	{{ 11,	15,	6},	{4,10,11,8,2,1}},
6	{{ 9,	14,	7},	{7,10,9,3,0,1}},
7	{{ 10,	14,	7},	{6,0,1,2,8,11,10,9}},
8	{{ 11,	14,	7},	{7,1,2,5,11,10}},
9	{{ 9,	15,	7},	{10,4,3,0,6,7}},
10	{{ 10,	15,	7},	{9,6,7,8,11,5,4,3}},
11	{{ 11,	15,	7},	{10,7,8,2,5,4}}}

Determining Latitude

The border conditions

$$\theta_0 \equiv \theta_{\text{north}} = 0$$
$$\theta_{11} \equiv \theta_{\text{south}} = \pi$$

and the average properties

$$4\theta_1 = \theta_0 + \theta_4 + \theta_2 + \theta_7$$
$$3\theta_2 = \theta_1 + \theta_5 + \theta_8$$
$$3\theta_3 = \theta_4 + \theta_0 + \theta_9$$
$$4\theta_4 = \theta_3 + \theta_{10} + \theta_5 + \theta_1$$
$$\vdots$$
$$4\theta_{10} = \theta_9 + \theta_7 + \theta_{11} + \theta_4$$

can be arranged in matrix notation:

$$\begin{pmatrix}
1 & 0 & & 0 & & & 0 & & & & & & \\
-1 & 4 & -1 & & -1 & & & -1 & & & & & \\
 & -1 & 3 & & & -1 & & & -1 & & & & \\
-1 & & & 3 & -1 & & & & & -1 & & & \\
 & -1 & & -1 & 4 & -1 & & & & & -1 & & \\
 & & -1 & & -1 & 3 & & & & & & -1 & \\
-1 & & & & & & 3 & -1 & & -1 & & & \\
 & -1 & & & & & -1 & 4 & -1 & & -1 & & \\
 & & -1 & & & & & -1 & 3 & & & -1 & \\
 & & & -1 & & & -1 & & & 3 & -1 & & \\
 & & & & -1 & & & -1 & & -1 & 4 & -1 & \\
 & & & & & 0 & & & 0 & & 0 & 1 &
\end{pmatrix}
\begin{pmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \\ \theta_5 \\ \theta_6 \\ \theta_7 \\ \theta_8 \\ \theta_9 \\ \theta_{10} \\ \theta_{11} \end{pmatrix}
=
\begin{pmatrix} \theta_{\text{north}} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \theta_{\text{south}} \end{pmatrix}. \quad (8)$$

Applying the first and the last row (boundary conditions) leads to the reduced, symmetric system

$$\begin{pmatrix} 4 & -1 & & -1 & & & -1 & & & \\ -1 & 3 & & & -1 & & & -1 & & \\ & & 3 & -1 & & & & & -1 & \\ -1 & & -1 & 4 & -1 & & & & & -1 \\ & -1 & & -1 & 3 & & & & & \\ & & & & & 3 & -1 & & -1 & \\ -1 & & & & & -1 & 4 & -1 & & -1 \\ & -1 & & & & & -1 & 3 & & \\ & & -1 & & & -1 & & & 3 & -1 \\ & & & -1 & & & -1 & & -1 & 4 \end{pmatrix} \begin{pmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \\ \theta_5 \\ \theta_6 \\ \theta_7 \\ \theta_8 \\ \theta_9 \\ \theta_{10} \end{pmatrix} = \begin{pmatrix} \theta_{\text{north}} \\ 0 \\ \theta_{\text{north}} \\ 0 \\ \theta_{\text{south}} \\ \theta_{\text{north}} \\ 0 \\ \theta_{\text{south}} \\ 0 \\ \theta_{\text{south}} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \pi \\ 0 \\ 0 \\ \pi \\ 0 \\ \pi \end{pmatrix} \tag{9}$$

The solution is $\boldsymbol{\theta} = (2\pi/5,\ 3\pi/5,\ 3\pi/10,\ \pi/2,\ 7\pi/10,\ 3\pi/10,\ \pi/2,\ 7\pi/10,\ 2\pi/5,\ 3\pi/5)^{\mathrm{T}}$, or $\boldsymbol{\theta}' = (0,\ 2\pi/5,\ 3\pi/5,\ 3\pi/10,\ \pi/2,\ 7\pi/10,\ 3\pi/10,\ \pi/2,\ 7\pi/10,\ 2\pi/5,\ 3\pi/5,\ \pi)^{\mathrm{T}}$, where the first and last value correspond to the poles.

Determining Longitude

In Fig. A2, the path 0, 1, 2, 5, 11 is used as the date line; it is indicated by a row of small black triangles. Links extending from the date line to the west are $1 \rightarrow 4$ and $5 \rightarrow 4$; they are marked with white triangles. The poles, "N" (vertex 0) and "S" (vertex 11), are no longer part of the net. The following equations result:

$$\begin{aligned} 3\phi_1 &= (\phi_4 - 2\pi) + \phi_2 + \phi_7 \\ 3\phi_2 &= \phi_1 + \phi_5 + \phi_8 \\ 2\phi_3 &= \phi_4 + \phi_9 \\ 4\phi_4 &= \phi_3 + \phi_{10} + (\phi_5 + 2\pi) + (\sigma_1 + 2\pi) \\ &\vdots \\ 3\sigma_{10} &= \phi_9 + \phi_7 + \phi_4. \end{aligned}$$

These equations, together with $2\phi_1 = 0$, can be put into matrix notation as

$$\begin{pmatrix} 5 & -1 & & -1 & & & -1 & & & \\ -1 & 3 & & & -1 & & & -1 & & \\ & & 2 & -1 & & & & & -1 & \\ -1 & & -1 & 4 & -1 & & & & & -1 \\ & -1 & & -1 & 2 & & & & & \\ & & & & & 2 & -1 & & -1 & \\ -1 & & & & & -1 & 4 & -1 & & -1 \\ & -1 & & & & & -1 & 2 & & \\ & & -1 & & & -1 & & & 3 & -1 \\ & & & -1 & & & -1 & & -1 & 3 \end{pmatrix} \begin{pmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \\ \sigma_4 \\ \sigma_5 \\ \sigma_6 \\ \sigma_7 \\ \sigma_8 \\ \sigma_9 \\ \sigma_{10} \end{pmatrix} = \begin{pmatrix} -2\pi \\ -0 \\ 0 \\ 4\pi \\ -2\pi \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}. \tag{10}$$

The solution is $\phi = (0,\ 0,\ 5\pi/4,\ 3\pi/2,\ -\pi/4,\ 3\pi/4,\ \pi/2,\ \pi/4,\ \pi,\ \pi)^{\mathrm{T}}$.

APPENDIX B: SPHERICAL HARMONIC FUNCTIONS

The following definitions are used in this paper [12].

Legendre Polynomials.

$$P_l(x) = \frac{1}{2^l l!}\frac{d^l}{dx^l}(x^2 - 1)^l. \tag{11}$$

Associated Legendre Polynomials.

$$P_l^m(x) = (-1)^m(1 - x^2)^{m/2}\frac{d^m}{dx^m}P_l(x) \tag{12}$$

$$= \frac{(-1)^m}{2^l l!}(1 - x^2)^{m/2}\frac{d^{m+l}}{dx^{m+l}}(x^2 - 1)^l.$$

Spherical Harmonic Functions.

$$Y_l^m(\theta, \phi) = \sqrt{\frac{2l + 1}{4}\frac{(l - m)!}{(l + m)!}}P_l^m(\cos\theta)\, e^{im\phi} \tag{13}$$

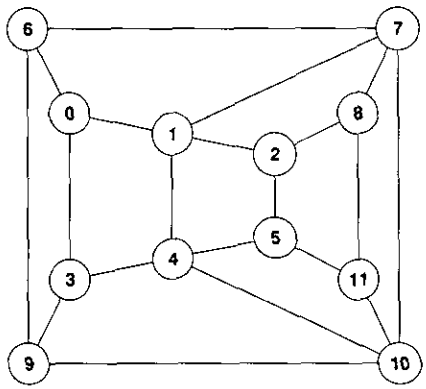

FIGURE A1

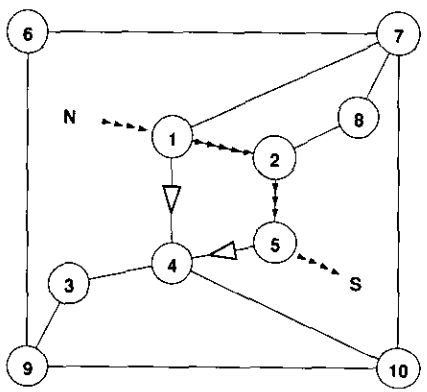

FIGURE A2

$$Y_l^{-m}(\theta, \phi) = (-1)^m Y_l^{m*}(\theta, \phi). \tag{14}$$

A spherical harmonic function of degree l can be written as a homogeneous polynomial of degree l in $u = \sin\theta \cos\phi$, $v = \sin\theta \sin\phi$ and $w = \cos\theta$.

REFERENCES

1. D. H. Ballard and Ch. M. Brown, *Computer Vision*, Prentice-Hall, Englewood Cliffs, NJ, 1981.
2. F. Solina and R. Bajcsy, Recovery of parametric models from range images: The case for superquadrics with global deformations, *IEEE Trans. Pattern Anal. Machine Intell.* **12**(2), 1990, 131–147.
3. L. H. Staib and J. S. Duncan, Deformable fourier models for surface finding in 3d images, *Visualization in Biomedical Computing 1992* (R. A. Robb, Ed.), Vol. Proc. SPIE 1808, pp. 90–104, 1992.
4. E. Persoon and K. S. Fu, Shape discrimination using fourier descriptors, *IEEE Trans. Sys. Man Cybernetics* **7**(3), 1977, 388–397.
5. F. P. Kuhl and Ch. R. Giardina, Elliptic fourier features of a closed contour, *Comput. Graphics Image Process.* **18**(3), 1982, 236–258.
6. D. N. Kennedy, J. Sacks, P. A. Filipek, and V. S. Caviness, Three-dimensional fourier shape analysis in magnetic resonance imaging, in *Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Philadelphia, 1990* (P. C. Pedersen and B. Onaral, Eds.), Vol. 12/1, pp. 78–79.
7. Ch. Brechbühler, G. Gerig, and O. Kübler, Surface parameterization and shape description, In *Visualization in Biomedical Computing 1992* (R. A. Robb, Ed.), Vol. Proc. SPIE 1808, pp. 80–89, 1992.
8. G. T. Herman and H. K. Liu, Three-dimensional display of human organs from computer tomograms, *Comput. Graphics Image Process.* **9**(1), 1979, 1–21.
9. D. Gordon and J. K. Udupa, Fast surface tracking in three-dimensional binary images, *Comput. Vision Graphics Image Process.* **45**(2), 1989, 196–214.
10. J. K. Udupa and V. G. Ajjangadde, Boundary and object labelling in three-dimensional images, *Comput. Vision Graphics Image Process.* **51**(3), 1990, 355–369.
11. Ch. Brechbühler, G. Gerig, and O. Kübler, Towards representation of 3-D shape: Global surface parametrization, in *Visual Form: Analysis and Recognition* (C. Arcelli, L. P. Cordella, and G. Sannití di Baya, Eds.), pp. 79–88, Plenum Press, New York/London, 1992.
12. W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. Vetterling, *Numerical Recipes in C—The Art of Scientific Computing*, Cambridge Univ. Press, Cambridge, 1988.
13. C. Pommerell and W. Fichtner, PILS: An iterative linear solver package for ill-conditioned systems, in *Proceedings of the Supercomputing '91, Albuquerque, New Mexico, November 18–22, 1991*, pp. 588–599.
14. L. Nirenbarg, A strong maximum principle for parabolic equations, *Commun. Pure Appl. Math.* **6**, 1953, 167–177.
15. P. E. Gill, W. Murray, and M. H. Wright, *Prictical Optimization*, Academic Press, London, 1981.
16. W. Greiner and H. Diehl, *Theoretische Physik—Ein Lehr- und Übungsbuch für Anfangssemester, Vol. 3. Elektrodynamik*, Verlag Harri Deutsch, Zürich/Frankfurt am Main, 1986.
17. P. T. Sander and S. W. Zucker, Inferring surface trace and differential structure from 3-d images, *IEEE Trans. Pattern Anal. Mach. Intell.* **12**(9), 1990, 833–854.