A General Framework for Implicit Modeling

Xikun Liang and Brian Wyvill
Department of Computer Science, University of Calgary
2500 University Dr. NW, Calgary, Canada T2N 1N4
Email:{ xkliang | blob } @cpsc.ucalgary.ca

Abstract

A new implicit modeling method is introduced which represents a complex object as a hierarchy of implicit surfaces, in which each level is constrained by its parent. The method can be used in a variety of applications in modeling and animation. Novel aspects of the method are its ability to progressively refine the model, to create complex implicit objects through the composition of blending, controlled blending, CSG Boolean operations, and precise contact modeling, and to change the overall shape of the model while maintaining the integrity of the object.

Most implicit modeling techniques build models that are collections of globally defined primitives or hierarchies in which each level is unconstrained by its parent. Our method builds a hierarchy of implicit surfaces progressively as new local details are introduced. As in H-Splines, local details are added to constrained areas and maintained during large scale shape changes. The method is well suited for both interactive and procedural implicit modeling. In addition, we describe a new methodology for the composition of implicit surfaces. We illustrate the intuitiveness and effectiveness of the new method in a number of examples of biological and engineering modeling and animation.

CR Categories: 1.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Hierarchy and geometric transformation; Curve, surface, solid and object representations.

Keywords: Implicit modeling, hierarchical surfaces, local refinement, global deformation.

1 Introduction

Parametric methods provide smooth curved surfaces but the designer is faced with the difficulty of managing large numbers of patches. Implicit models have some advantages, blending is automatic and methods have been developed for blending [Blinn 1982; Wyvill et al. 1986b; Bloomenthal 1997], polygonization [Wyvill et al. 1986b], ray tracing [Hart 1996; Wyvill and Trotman 1990; Kalra and Barr 1989], collision detection and process [Gascuel 1993], and animation [Wyvill et al. 1986a; Desbrun and Cani-Gascuel 1998; Gascuel 1993]. Implicit model is natural representation of a solid particularly when blending between primitives is required. Recent research has demonstrated the variety of applications of implicit surfaces [Bloomenthal 1997; Markosian et al. 1999; Frisken et al. 2000].

Despite all the implicit modeling techniques developed so far, building an implicit model with local details is still a problem, both for interactive and procedural modeling. Even hierarchical modeling techniques such as the *BlobTree* [Wyvill et al. 1999], do not allow a parent level to constrain the child nodes in the hierarchy. This means that once a model is defined, any change to part of the model may result in unexpected changes to the whole model.

Modeling with implicit surfaces guarantees a continuous smoothly blended surface except when a constructive solid geometry (CSG) operation [Ricci 1973] is involved. Users often experience difficulties with implicit modeling, as it is not easy to predict the precise shape of the surface resulting from the combination of primitives using operations such as blending. They often want to make further refinement to a surface without altering the gross shape or non local details. Motivated by hierarchical B-Splines [Forsey and Bartels 1988], the new method presented in this paper tackles the problem by building a hierarchy of implicit surfaces with both local refinement and global deformation embedded.

Building on the pioneering work in combining CSG, and warping with blending [Wyvill et al. 1999], we introduce new refinement techniques inspired by the work of Forsey [Forsey and Bartels 1988]. The objective is to develop a practical geometric implicit modeling technique and to make it easy for a designer to build complex models, and edit the models by refining the details.

The hierarchical representation of implicit surfaces presented in this paper is suitable for both interactive and procedural modeling. The structure described in this paper can also be used to facilitate the animation of complex implicit models, since local details and global deformations can be easily controlled and changed over time.

Our research has focused on skeletal implicit surfaces S, which is characterized as a collection of points, M in space with a potential $F(x,y,z)$ equal to some threshold value denoted by T.

$$S = \{M(x,y,z) \in \mathscr{R}^3 \mid F(x,y,z) = T\} \quad (1)$$

1.1 Related work

Implicit surfaces have been successfully used in the design of smooth objects of arbitrary topology and geometry [Bloomenthal 1997; Stander and Hart 1998; Bloomenthal and Shoemake 1991], as well as in descriptive and physically based animation systems [Bloomenthal 1997; Wyvill et al. 1986a; Desbrun and Cani-Gascuel 1998; Gascuel 1993]. Particularly, the *BlobTree* [Wyvill et al. 1999] provides a hierarchical method which allows arbitrary composition of models that make use of blending, warping and Boolean operations. The *BlobTree* suffers the same problem as other implicit modeling methods in that it is difficult to change the model after it is created. The model fails even in a simple case as in Figure 1. An intuitive result would be like the model in Figure 1.d when the central column is bent. With current modeling technique the model (Figure 1.a) either collapses (Figure 1.b) or produces weird shape (Figure 1.c) under a shape change of part of the model. The problem is that the parent node does not contain information to constrain its children.

Liang et al. [Liang and Wyvill 2001] presented an initial solution to this problem by introducing some constraint based hierarchical techniques. In this paper we improve and expand on these techniques, introducing several new techniques and improving many of the earlier methods. We describe our new hierarchical refinement methods as applied to implicit modeling, including blending, warping, CSG, and Precise Contact Modeling

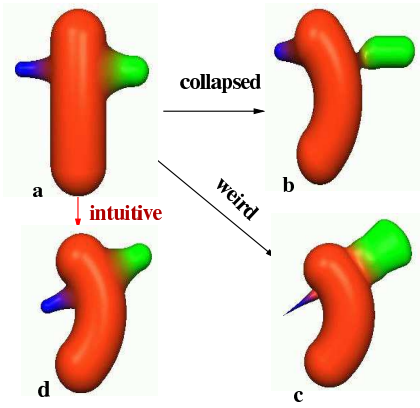

Figure 1: Global deformation fails for unconstrained hierarchical implicit models: a. a cross model built from three line primitives, b. collapsed after deformation of the central cylinder, c. weird shape produced after deformation of the whole model, d. intuitive deformed shape

(PCM) [Gascuel 1993]. The major contribution over simply combining these methods in the *BlobTree* is that the refinement techniques allow a designer to introduce a constrained volumetric representation with a new group of primitives and to change the overall shape of the object without having to re-edit local surface details.

1.2 Overview

The new method referred to as *HISR*, Hierarchical Implicit Surface Refinement, builds a model in a hierarchy of implicit surfaces. Each *HISR* contains a global surface and a set of sub-*HISRs* which are defined relative to the global surface. Changes to the global surface only affects surfaces in its sub-*HISRs* while changes to a surface in the sub-*HISRs* only affects surfaces under it in the structure. Designers start with a basic surface and progressively introduce new surfaces as local details of an existing surface. Local constraints combined with the hierarchy are used to control the shape of the model. Hierarchical operations such as affine transformations and field warping operations are designed to be applied to any surface in the hierarchy for changing the overall shape of the surface. As in [Forsey and Bartels 1988], surface details are maintained intuitively during global deformation. The object represented by a *HISR* is obtained by constraint based hierarchical composition of all surfaces in the *HISR* using blending, controlled blending, CSG and PCM. Applications in biological modeling and animation, and engineering modeling show the effectiveness of the method.

The paper is organized as follows. In section 2, we present the general framework for the hierarchical representation of implicit surfaces and define four different types of hierarchies for the composition of implicit surfaces using blending, controlled blending, CSG Boolean operations, and precise contact modeling. Section 3 defines the hierarchical operations for local and global deformations. Section 4 presents a variety of applications of *HISR* in interactive modeling, biological and engineering modeling and animations. Our conclusions and future work are presented in Section 5.

2 A General Framework for Implicit Modeling

An intuitive implicit modeling method is to build objects in a hierarchy of implicit surfaces. Such a hierarchy supports both local surface refinements and global deformations. Local details can be added arbitrarily to subsections of the surface and global deformations can be performed to change the overall shape of the surface while maintaining the surface integrity.

Implicit surfaces are receiving increased attention especially with respect to their natural representation of solid objects and their innate blending properties. An implicit surface defined by a skeleton provides both articulation and realism and can facilitate animation, procedural definition, developmental processes, and replication with alterations. However, current implicit modeling techniques cannot effectively control the blended implicit surfaces. Primitives are simply combined all together, making it difficult to control the primitives individually. Although the *BlobTree* has the ability to model individual primitives, it is impossible to make changes to them after the model is built as illustrated in Figure 1. We develop a hierarchical representation of the implicit surfaces, with which a hierarchy of the details can be established and is accessible at any time.

2.1 Definition of the HISR

Definition 2.1 *A* HISR *is defined as a hierarchy of implicit surfaces, incorporated with operations of hierarchical local refinement and global deformation. It consists of a global surface and a set of sub-*HISRs.

Definition 2.2 *A* global surface *is an implicit surface created based on Equation 1.*

A global surface could be any implicit surface such as a single primitive, a transformed primitive surface, or a complex implicit surface built from a set of primitives through blending, controlled bending, CSG, and warping operations.

Definition 2.3 *A* sub-*HISR is a* HISR *which is defined in its local coordinate system relative to the coordinate system of its parent* HISR. *It represents a local surface detail of its parent* HISR.

A sub-*HISR* is local to its parent *HISR* and is transformed to a local frame in its parent *HISR*. This local frame is determined by the constraints applied to the sub-*HISR*, for instance the center of interest and orientation. It is automatically adjusted during any change of the parent *HISR* .

Definition 2.4 *A* combination *is a* HISR *operator which combines the global surface and surfaces of sub-*HISRs *according to specific rules.*

Definition 2.5 *An implicit surface represented by a* HISR *is defined as the hierarchical combination of a global surface and a set of surfaces represented by its sub-*HISRs. *The global surface itself is the surface represented by a leaf* HISR.

Based on the skeletal implicit surfaces, a surface represented by a hierarchy at level i, S^i, can be expressed as :

$$\begin{aligned} SH^i &= S^i_{global} \oplus \Sigma_k^n SH^{i+1}_{local_k} \\ &= \{M(x,y,z) \in \mathscr{R}^3 \mid F(x,y,z) = T\} \end{aligned} \quad (2)$$

where $\oplus$ refers to the combination operation depending on the type of *HISR*. S^i_{global} is the global surface of the hierarchy and $SH^{i+1}_{local_k}$ is the surface represented by its k'th sub-hierarchy. $n(\geq 0)$ is the number of sub-*HISRs*. $F(x,y,z)$ is the field function of the implicit surface SH^i.

2.2 Hierarchical Representation of HISR

According to the definition above, each *HISR* contains an implicit surface, either local or global, and a set of sub-surfaces from its sub-*HISRs*. A surface is local relative to its parent *HISR* and global relative to its sub-*HISRs*. A sub-*HISR* is defined in its own local coordinate system and represents a local detail of the global surface at its parent *HISR*. A local surface is transformed to a local frame in its parent coordinate system. The leaf node of the hierarchy is referenced based on its parent coordinate system.

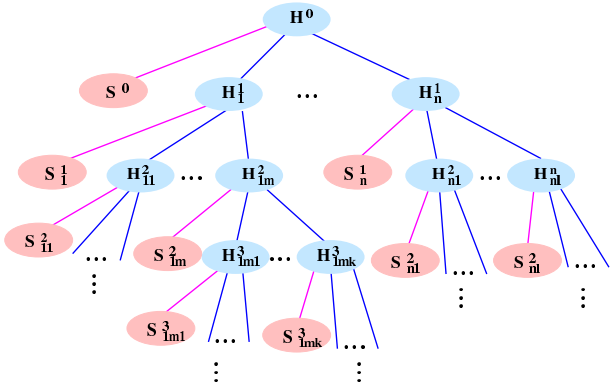

Figure 2: Hierarchy of *HISRs*

Figure 2 shows the hierarchical structure of *HISRs* where the blue (light) nodes are *HISRs* and pink (dark) nodes are global or local surfaces. "$\cdots$" indicates that there can be a number of *HISRs* in between. H^0 is the root *HISR*. S^0 is the global surface of the root *HISR* and $H_1^1, \cdots, H_n^1$ are the sub-*HISRs* of H^0. Similarly, S_1^1 is the global surface of an internal *HISR* H_1^1 at level 1 and $H_{11}^2, \cdots, H_{1m}^2$ are the sub-*HISRs* of H_1^1.

By definition, S^0 is a global surface for all other surfaces in the hierarchy. Surfaces $S_1^1, S_2^1, \cdots, S_n^1$ are local details on S^0 and are global to their sub-surfaces. For instance, S_1^1 is a global surface for surfaces $S_{11}^2, S_{12}^2, \cdots, S_{1m}^2$

The implicit surface represented at a level-*i HISR*, $SH^i_{\ldots j}$, consists of a global surface $S^i_{\ldots j}$ and a set of local surface details from its sub-*HISRs* $\{SH^{i+1}_{\ldots j_1}, \ldots, SH^{i+1}_{\ldots j_n}\}$, in which n is the number of the sub-*HISRs*. Such a surface is obtained by:

$$\begin{aligned} SH^i_{\ldots j} &= S^i_{\ldots j} \oplus \textstyle\sum_k^n SH^{i+1}_{\ldots j_k} \\ &= \{M(x,y,z) \in \mathscr{R}^3 \mid F(x,y,z) = T\} \end{aligned} \tag{3}$$

We define two types of operations, local refinement and global deformation. These are described below.

For a local refinement operation, an implicit surface is first defined in its own object coordinate system. A surface in the existing *HISR* is selected as the global surface to be refined. A surface point on this global surface defines the center of interest for the local detail to be introduced. The orientation of the local surface is determined by a local frame obtained by the tangent plane and the surface normal at the center of interest on the global surface. Constraints such as the offset and divergence of the local frame can be specified as required. Such a local refinement operation will only affect the selected global surface and the corresponding *HISR*. Other constraints can be further used to finely control the local details.

In a global deformation operation, a surface is chosen from the *HISR* first. It is then deformed, either by warping or transformation, as requested. Since the sub-*HISRs* of this *HISR* are defined based on the global surface, local frames of these sub-*HISRs* need to be re-evaluated accordingly in order to maintain the integrity of local details. This is done automatically while the global deformation is applied.

2.3 The HISRs

To obtain fine shape control of the implicit surfaces, four main kinds of *HISRs* are designed based on the most useful implicit modeling techniques that we have found from Wyvill's work [Wyvill et al. 1999; Gascuel 1993]. They represent four different *HISR* operations to combine a global surface with its local details. *HBlend* uses the super-elliptic blending method to create a smooth surface blended from the global surface and local details. *HCBlend* controls blending properties among local surface details [Guy and Wyvill 1995; Desbrun et al. 1996]. *HCSG* performs CSG operation [Ricci 1973], i.e. intersection, difference, or union, between the global surface and its local details. *HPCM* combines the global surface and the local details by creating exact contact surfaces between them using the method proposed in [Gascuel 1993].

2.3.1 HBlend - Blending HISR hierarchy

Definition 2.6 HBlend *contains a global surface and a set of sub-*HISRs*. It combines the global surface and surfaces represented by its sub-*HISRs *using super-elliptic blending to produce a smoothly blended surface.*

The surface represented by *HBlend* is obtained by:

$$SH_{HBlend} = S_{global} \oplus \sum_k^n SH_{local_k} \tag{4}$$

Given a space point M, the field value of *HBlend* is obtained by summing the field values of the global surface and the field values of all sub-*HISRs* :

$$F_{HBlend}(M) = F_{global}(M) + \sum_k^n F_{local_k}(M) \tag{5}$$

Although super-elliptic blending is used for the combination of global surface and local details in an *HBlend*, other blending methods are equally applicable. In fact, any blending function which produces a smoothly blended surface could be used.

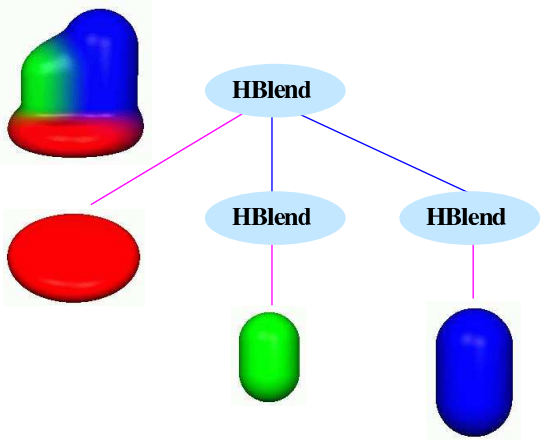

Figure 3: An object built from *HBlend*

Figure 3 shows an example of an *HBlend* object in which a global surface, a disk, is smoothly blended together with its two local

details from its sub-*HBlend*s. Notice that the two local surfaces are smoothly blended as well.

2.3.2 HCBlend - Controlled blending HISR hierarchy

Definition 2.7 HCBlend *consists of a global surface and a number of sub-*HISRs. *It combines its global surface and surfaces represented by its sub-*HISRs *using controlled blending techniques. All surfaces from the sub-*HISRs *are blended with the global surface, but control blended with each other.*

The surface represented by an *HCBlend* is obtained by:

$$SH_{HCBlend} = S_{global} \oplus \sum_{k}^{n} SH_{local_k} \tag{6}$$

We have chosen to use the controlled blending technique from Guy [Guy and Wyvill 1995] because of its simplicity. Desbrun's method [Desbrun et al. 1996] is equally applicable.

Given a space point M, the field value of an *HCBlend* is obtained by summing the field value of the global surface and the controlled field values of all sub-*HISRs* :

$$F^i_{HCBlend} = F^i_{global} + \sum_{k}^{n} ct(F_{local_k}(M)) \tag{7}$$

where $ct(F_{local_k}(M))$ is the controlled field value from the local details as in [Guy and Wyvill 1995].

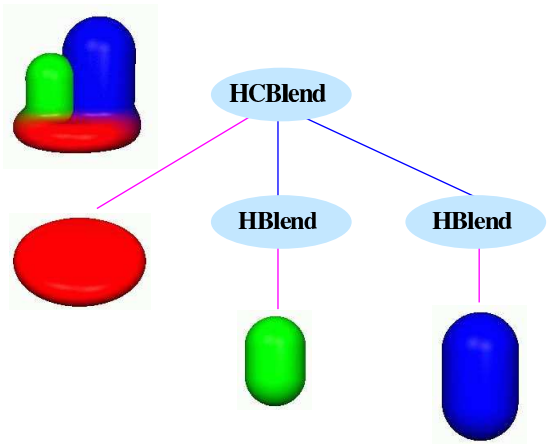

Figure 4: An object built from *HCBlend*

Figure 4 shows an example of an *HCBlend* object in which a global surface, a disk, is smoothly blended together with its two local details. The two local surfaces (vertical cylinders) are not blended as defined by the controlled blend operation.

2.3.3 HCSG - CSG HISR hierarchy

Definition 2.8 HCSG *contains a global surface and a set of sub-*HISRs *. It deals with refinement related to CSG Boolean operations such as intersection, difference, and union. It combines its global surface and surfaces represented by its sub-*HISRs *by applying the CSG Boolean operation sequentially through the list of sub-*HISRs.

HCSG represents implicit solid modeling in *HISR*. It consists of *HUnion*, *HIntersection*, and *HDifference*, representing CSG Boolean operations of union ($\cup$), intersection ($\cap$) and difference ($-$) respectively [Ricci 1973]. The surface of an *HCSG* represents a solid as the set of points where an implicit global defining function takes on a value less than a given threshold value. Global surface and surfaces from the sub-*HISRs* are combined using a tree structure as in constructive solid modeling. Combination operations are implemented by functions which provide an implicit defining function for the resulting solid. An *HCSG* represents the following surface:

$$SH_{HCSG} = (\cdots(S_{global} \left\{ \begin{array}{c} \cup \\ \cap \\ - \end{array} \right\} SH_{local_1}) \cdots \left\{ \begin{array}{c} \cup \\ \cap \\ - \end{array} \right\} SH_{local_n}) \tag{8}$$

Given a space point M, the field value of an *HCSG* is obtained by:

$$\begin{array}{lcl} F_{\cup}(M) & = & max(\cdots(max(F_{global}, F_{local_1}), \cdots, F_{local_n}) \\ F_{\cap}(M) & = & min(\cdots(min(F_{global}, F_{local_1}), \cdots, F_{local_n}) \\ F_{-}(M) & = & max(\cdots(max(F_{global}, 2*T - F_{local_1}), \cdots, 2*T - F_{local_n}) \end{array} \tag{9}$$

where $max(x,y)$ and $min(x,y)$ are functions to find the maximum and minimum of x and y. T is the threshold value for the implicit surface.

Figure 5 shows examples of different *HCSG* objects in which a global surface, a sphere, is combined together with its one or two local details, spikes or sphere, from its sub-*HBlend*s.

2.3.4 HPCM - Precise Contact Modeling HISR hierarchy

Definition 2.9 HPCM *is composed of a global surface and a set of sub-*HISRs *representing surface details. It defines separate surfaces which are in precise contact.*

An *HPCM* creates $n+1$ separate objects where n is the number of the local details. A precise contact surface is generated between any two surfaces, either global and local detail or local detail and local detail, if they are in collision.

The surface represented by an *HPCM* is defined as follows:

- The main surface created from the global surface:

$$SH_{HPCM} = S_{global} \oplus \sum_{k}^{n} SH_{local_k} \tag{10}$$

- Sub-surfaces represented by a sub-*HISR* SH_{sub_r} $(1 \le r \le n)$:

$$SH_{sub_r} = SH_{local_r} \oplus S_{global} \oplus \sum_{k,k \neq r}^{n} SH_{local_k} \tag{11}$$

As in [Cani-Gascuel 1998], the contact surface is modeled by considering both the inter-penetration region and the propagation region of the contact surfaces.

Given a space point M, the field values of an *HPCM* is obtained as follows:

$$\begin{array}{lcl} F_{HPCM}(M) & = & F_{global}(M) + g_{global}(M) \\ F_{sub_r}(M) & = & F_{local_r}(M) + g_{local_r}(M) \end{array} \tag{12}$$

where $F_{HPCM}(M)$ and $F_{sub_r}(M)$ are the field functions of the resulting surfaces for the global surface and the r'th local surface respectively. $F_{global}(M)$, $F_{local_r}(M)$, $g_{global}(M)$, and $g_{local_r}(M)$ are the field functions and the deformation fields for the global surface and the r'th sub-surface respectively. $g_{global}(M)$ and $g_{local_r}(M)$ are defined as follows:

$$g_{global}(M) = \left\{ \begin{array}{ll} \sum_k^n (T - F_{local_k}(M)), & F_{global} \ge T \\ & \wedge \\ & \forall k \, F_{local_k} \ge T \\ \sum_k^n g_{global_prop_k}(M), & otherwise \end{array} \right. \tag{13}$$

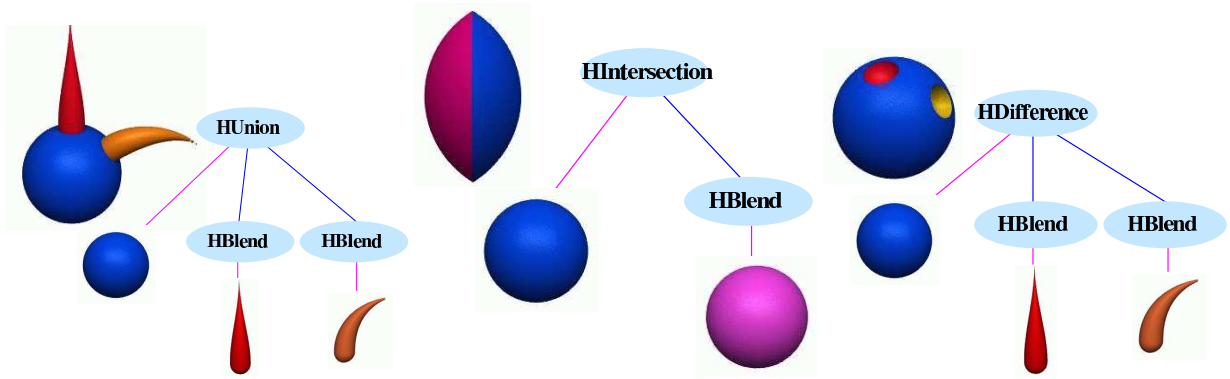

Figure 5: Objects built from *HUnion* , *HIntersection* and *HDifference*

$$g_{local_r}(M) = \begin{cases} T - F_{global}(M) \\ +\sum_{k,k\neq r}^{n}(T - F_{local_k}(M)) \ , & F_{local_r} \geq T \\ & \wedge \\ & F_{global} \geq T \\ & and\ \forall k \\ & F_{local_k} \geq T \\ g_{global_prop(M)} \\ +\sum_{k,k\neq r}^{n} g_{local_prop_k}(M) \ , & otherwise \end{cases} \tag{14}$$

where $g_{global_prop_k}(M)$ is the dilation field value caused by the k'th local surface to the global surface, $g_{global_prop}(M)$ by the global surface to the r'th local surface, and $g_{local_prop_k}(M)$ by the k'th local surface to the r'th local surface. These functions are defined by decreasing functions in the specified range of dilation as in [Gascuel 1993].

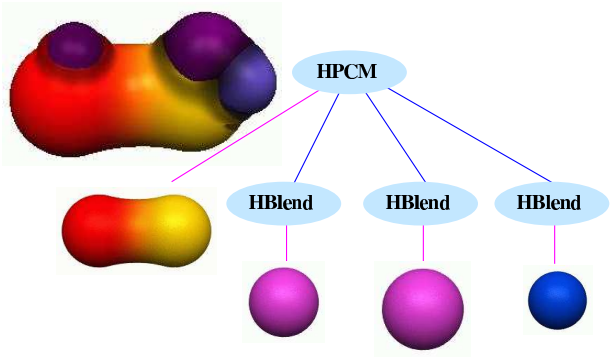

Figure 6: An example of *HPCM HISR*

Figure 6 shows an example of an *HPCM* object with three sub-*HISRs*. The global surface, a peanut built by two blended spheres, is combined together with its three local details, three different sized and colored spheres, from its sub-*HBlend*s. The exact contact surfaces are modeled between all contact surfaces, not only global and local, but local and local spheres as well.

2.4 Building the HISR Model

A *HISR* model can be built procedurally, interactively, or both. With an initial implicit surface, a root *HISR* is first created. Then a sequence of sub-*HISRs* can be built as local details of selected global surfaces in the *HISR*. This procedure continues until all the details have been added.

To introduce a local detail to the *HISR*, a description of the detail is specified first, for instance using a specific primitive, a pre-defined surface, or a specified volume. A level-i sub-*HISR* H^{i+1} is created with a new surface S^{i+1}, determined by a splitting scheme $split()$ which searches the *HISR* , finds related surfaces according to refinement constraints (cf. section 3), and generates a set of new primitives $\{p_k\}$ at level-$(i+1)$.

$$\begin{array}{lcl} S^{i+1} & = & \{M(x,y,z) \in \mathscr{R}^3 \mid F^{i+1}(x,y,z) = T\} \\ F^{i+1}(x,y,z) & = & \sum_k F_k(x,y,z) \mid p_k \in split(S^i_{surface}) \end{array} \tag{15}$$

Each *HISR* is constructed in its own local coordinate system which is then transformed to a local frame in its parent coordinate system (world coordinate system for the root hierarchy). The local frame is determined by the center of interest of the refinement area O, the normal vector of the center, and constraints (refer to Figure 8), i.e.:

$$\begin{cases} Y & := & (\partial F/\partial x, \partial F/\partial y, \partial F/\partial z)(O) \\ X & := & (\partial F/\partial x, 0, 0)(O) \times M_{N \leftarrow Y_p} \\ Z & := & X \times Y \end{cases} \tag{16}$$

where O is a point on the parent surface in the parent coordinate system and F is the field function of the parent surface. Y is the normal vector at O on the parent surface. X is a tangent vector at O obtained by rotating the unit vector along the parent frame's X axis by a rotation matrix $M_{N \leftarrow Y_p}$. This makes sure that all local frames at a certain level of the *HISR* are constrained by their parent coordinate system.

With this *HISR*, the final object is obtained by combining all surfaces below or at the same level using a bottom-up surface construction strategy. Lower level *HISRs* are processed first. Then they are combined with higher level surfaces constrained by the type of the *HISR* and other specified constraints. Local details are maintained based on refinement constraints and hierarchical operations.

2.5 Visualization of the HISR Model

A *HISR* represents an implicit surface or a list of implicit surfaces through the hierarchical composition of the global surfaces and constraints. To visualize the implicit surface of such a *HISR*, polygonization or ray-tracing algorithms needs to evaluate the implicit field function at a large number of points in space.

Given a node N in the *HISR* hierarchy, the following algorithm returns a field value of a space point M for the main object built from the global surface of the root *HISR*. Field values of other surfaces can be obtained similarly.

Function $F(N,M)$:

1. if N is a Leaf node: F_{global}
2. if N is *HBlend* hierarchy: $F_{global} + \sum_k^n F_{local}$
3. if N is *HCBlend* hierarchy: $F_{global} + \sum_k^n ct(F_{local})$
4. if N is *HPCM* hierarchy: $F_{global} + g_{global}$
5. if N is *HCSG* hierarchy:
   (a) *HUnion* : $max(\cdots, max(F_{global}, F_{local_1}), \cdots, F_{local_n})$
   (b) *HIntersection* : $min(\cdots, min(F_{global}, F_{local_1}), \cdots, F_{local_n})$
   (c) *HDifference* : $min(\cdots, min(F_{global}, 2 * T - F_{local_1}), \cdots, 2*T - F_{local_n})$

Figure 7: Computing the field value of a given node N in *HISR*

Using a uniform space subdivision method [Wyvill et al. 1986b], a polygon mesh can be generated, which represents a surface built from the *HISR*. A set of field values need to be obtained at the subdivision grid points based on the function $F(N,M)$.

Another method is to render the *HISR* model directly. The basic process is to find pixel value for each ray corresponding to the pixel. This requires to search the ray-surface intersection and surface normal. The interval analysis based ray tracing of the *BlobTree* [Wyvill et al. 1999] could be applied to *HISR*. Let $r(t) = r_0 + \delta t$ be the parameterized ray. Intersection occur at points $r(t)$ such that $F \circ r(t) = T$. Interval analysis finds solutions by defining the function $F \circ r(t)$ and its derivative over intervals, and uses interval analysis to bound those functions [Snyder 1992]. Other implicit ray tracing methods proposed in [Bloomenthal 1997] are equally applicable.

3 Constraint-based Hierarchical Operations

3.1 Local Refinement

Local refinement of an implicit surface can be achieved by using hierarchy information, local constraints, and implicit surface properties such as the field function, radius of influence, etc.

3.1.1 Constraints

We define the following constraints to obtain the locality of refinement:

- **Point constraint**: A point constraint is defined by a point on the surface being refined. Local detail will be bounded to this point. Usually, this is the origin of the local frame for the local detail.
- **Local area constraint**: By limiting the range of influence of the primitives, refinement can be restricted within a specified region. Such a region on the surface defines a local area constraint (the shaded area in Figure 8). The radius of influence of the new primitive or set of primitives is the minimal distance between the center of the detail and the boundary of the local area or volume, i.e. where $F > 0$.
- **Level constraint**: As the object is represented in a hierarchy of implicit surfaces, the level of a surface in the *HISR* is used for local refinement as well. Given a specific level, local details will be added only on one of the global surfaces at that level.
- **Surface constraint**: A global surface can also be explicitly specified as the surface to be refined. In this case, subsequent local details will be added directly on this surface. This is the usual case for procedural modeling using our method.

3.1.2 Modeling Surface Details

Local details are introduced according to the refinement constraints. To add details into the hierarchy, we first specify the required constraint and a specific point on the surface, usually being the center of interest for the local refinement. The center of interest is used to traverse the hierarchy to obtain a group of global surfaces whose field values are greater than some threshold value. Depending on what constraint is defined, the surface to be refined are found as follows:

- *Point constraint*: The point itself is used as the center of interest for querying the hierarchy to obtain a surface in the *HISR* which has the greatest contribution (largest field value) on this point.
- *Local area constraint*: A point within the local area is specified as the center of interest. A surface with the greatest contribution to the center of interest is chosen from a group of surfaces as the surface to be refined.
- *Level constraint*: A surface with the greatest contribution at the given level is chosen as the surface to be refined.
- *Surface constraint*: The surface of the constraint automatically becomes the surface to be refined.

The selected surface and its corresponding hierarchy will be the parent surface and hierarchy for subsequent local refinements.

A new surface is constructed based on the constraints, a group of surfaces contributing to the refinement, and the surface description that defines the expected shape of the detail. Such a surface could be created using a single primitive, a complex surface built from a set of skeletons, or even a surface built from a *HISR*. It is defined in its own object coordinate system.

Creating the local surface is termed the *splitting scheme* in our framework. A simple splitting technique introduces a single new primitive on the parent surface S^i_{parent} for each refinement, while keeping other primitives of the parent surface unchanged, i.e.

$$S^{i+1}_{local} = split(S^i_{parent}) = \{S_p \mid p_{primitive}\} \tag{17}$$

where S_p is a surface defined by primitive $p_{primitive}$. The primitive could be a point primitive for a small bump or a tapered line primitive for a spike, etc. The properties of the primitive, such as the radius of influence and potential function, are determined based on the constraints. Figure 8 shows a simple case using this scheme. The refinement introduces a local spike centered at O. The characteristics of the spike are determined based on a local area constraint and some tapering and bending constraints.

Another splitting scheme uses a given surface as the local detail. More complicated surfaces can be used as local details, providing more flexibility of modeling complex objects.

$$S^{i+1}_{local} = split(S^i_{parent}) = Q \tag{18}$$

where Q is a pre-defined implicit surface.

These splitting methods provide interactive as well as satisfactory results for most modeling requests. However local details are

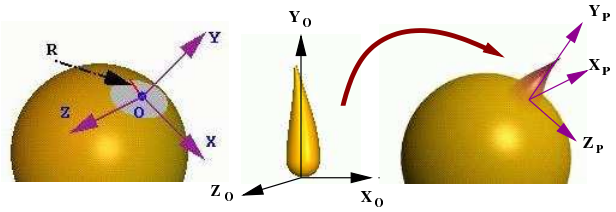

Figure 8: Surface refinement with a simple splitting scheme

limited by the availability of primitives and the primitives within the model may not be the optimal set of primitives that models the object. Redundant primitives may exist after a series of refinements. Therefore, a better scheme is needed to be investigated, in particular, a volume based splitting scheme may be used to obtain an optimal set of primitives approximating a local detail.

3.1.3 Adding local details into the HISR

With the new local detail, a new *HISR* is created and is added as a sub-*HISR* at the chosen parent surface hierarchy. It is transformed to a local frame of its parent coordinate system. Given the center of interest $O = [O_x, O_y, O_z, 1]^T$ in the parent coordinate system XYZ, the transformation, A (a 4×4 matrix) is determined by the following equation:

$$A = T(O_x, O_y, O_z) \cdot R_x(\phi) \cdot R_y(\theta) \tag{19}$$

where

$$\begin{cases} N &= (\partial F/\partial x, \partial F/\partial y, \partial F/\partial z)(O) \\ \theta &= \arctan(N_z/N_x) - 90 \\ \phi &= \arctan(N_y/\sqrt{N_z^2 + N_x^2}) \end{cases}$$

As a result, a surface in *HISR* is globally referenced as:

$$S = S_{local} * A_{parent} * A_{grand_parent} * \cdots * A_{root} \tag{20}$$

where A transforms a local surface to a higher level surface and A_{root} is an identity matrix.

3.2 Global Deformations

Global deformation refers to the deformation of any surface in *HISR*. Its main purpose is to change the overall shape of a surface at some specific layer while maintaining the consistency and integrity of its local details. Two commonly used types of global deformations, i.e. the affine transformations and field warping operations, are defined. The major issue involved in global deformation is how to automatically adjust the sub-surfaces without having to re-edit or re-define the details afterward. As each surface in the *HISR* is defined in a local frame relative to its parent coordinate system, these local frames of the sub-hierarchies can be adjusted so as to keep the consistency of the surfaces. In the following sections, assume that:

- X, Y, Z, X', Y', Z': the base vectors of the original XYZ and new $X'Y'Z'$ local frame,
- O, O': the origins of XYZ and $X'Y'Z'$ in its parent coordinate system,
- $T_{XYZ}, T_{X'Y'Z'}$: matrices to transform the local frame to its parent coordinate system,

where X is a normalized tangent vector of the surface at O and Y is the surface normal at O. All vectors are column vectors represented in homogeneous coordinate system.

3.2.1 Affine Transformations

Affine transformations are the most popular global deformations. They change the overall shape of the object through translation, rotation, scaling, or shearing. A 4×4 matrix $\mathbf{T}$ is made to represent an affine transformation or a composite transformation. Given a surface in the hierarchy and an expected transformation $\mathbf{T}$, the new local frame for any sub-hierarchy of the surface can be obtained as follows:

$$\begin{array}{lcl} O' &=& moveToSurface(\mathbf{T} \cdot O) \\ Y' &=& (\partial F/\partial x', \partial F/\partial y', \partial F/\partial z')(O') \\ T_{X'Y'Z'} &=& T_{X'Y'Z' \leftarrow XYZ} \cdot T_{XYZ} \end{array} \tag{21}$$

where function $moveToSurface()$ searches the transformed surface point O' on the parent surface with an initial point $\mathbf{T} \cdot O$ along the surface gradient direction. $T_{X'Y'Z' \leftarrow XYZ}$ is the matrix which transforms O to O' and aligns vector Y to vector Y' in its parent coordinate system.

3.2.2 Field Warping Operations

Field warping operations have been proven very useful for implicit modeling [Wyvill et al. 1999]. After warping is applied on the surface, the warped sub-*HISRs* are then defined in the following local frame:

$$\begin{array}{lcl} O' &=& \mathbf{warp}(O) \\ Y' &=& (\partial F/\partial x', \partial F/\partial y', \partial F/\partial z')(O') \\ T_{X'Y'Z'} &=& T_{X'Y'Z' \leftarrow XYZ} \cdot T_{XYZ} \end{array} \tag{22}$$

where $T_{X'Y'Z' \leftarrow XYZ}$ is the same as in affine transformations and **warp**() could be any warping operation such as bend, taper, twist, or user-defined functions.

3.2.3 Recursive Deformations

In addition to common affine transformations, a special class of affine transformations is designed for recursive operations, e.g. recursive scaling or shearing. The transformation is propagated recursively down the hierarchy. Such an operation is especially useful when all local details of a surface need to be properly deformed as well. In this case, equation 21 is first applied to the global surface and then to each of the surface in its sub-*HISRs*. This process continues until the leaf *HISR* is reached.

3.2.4 Limitations

The new local frame is determined based on the assumption that the deformed center of interest O' is defined and is a regular point on the deformed global surface. This makes sure that local details are still constrained to the same surface after global deformation. Otherwise, the integrity of the surface cannot be guaranteed to be maintained since there is no corresponding local frame on the deformed global surface. This usually happens when the center of interest of a local detail is either transformed or warped from a regular point on the original surface to a singular point on the deformed surface. Extreme deformations may cause the same problem.

4 Results

In this section, we demonstrate some applications of the proposed method. We first describe an interactive method and then show how to model a complex biological object, i.e. a sea anemone. An engineering model, distributor cap, shows the ability of the method to create solid geometric models.

4.1 Interactive Modeling

HISR is well suited for building models interactively. Local refinements allow a designer to add surface details at any level of the *HISR* model during the modeling process. Global deformations change the overall shape of the surface at a given hierarchy in which local details are automatically adjusted to maintain the integrity and consistency of the surface after the global deformations.

The cross model in Figure 1 can be very easily built interactively using *HISR*. A root *HBlend* hierarchy is first created as the column using a line primitive as its global surface. Two smaller line primitives are modeled as local details to the surface of the column by specifying a center of interest on the column and constraints for each of them. They are then added to the root hierarchy as sub-*HISRs* that are *HBlends* with the smaller line primitives as their global surfaces. A hierarchical bending operation is subsequently applied to the root to deform the cross model, producing the intuitive bent cross object as in Figure 1.d.

As shown in [Liang and Wyvill 2001], with a simple surface at the root *HISR*, a series of local refinements and global deformations can be used to build a complex model. This is especially effective for prototyping models.

4.2 Building a HISR Sea Anemone

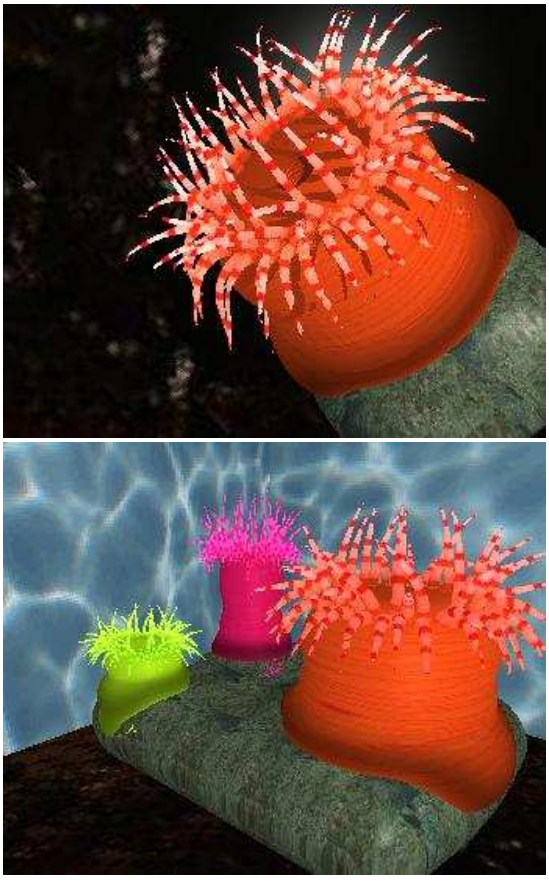

Figure 9: Sea anemone *HISR* models attached to a rock

Anemones, or the flower animals, consist of five main components, the base, the column, the upper disk, the mouth, and the tentacles, as described in [Stephenson 1934; Harbo 1949]. The mouth is a slit at the center of the upper disk, surrounded by a number of tentacles which are generally arranged in four or five cycles. The bottom of the tentacles are smoothly blended together with the other part of the body. Anemones change their shape and structure in response to various stimuli. For example, various body deformations such as contractions and extensions occur, and the tentacles' orientations, size and bending properties change [Elliot et al. 1989]. Their soft body and shape deformation behavior can be easily represented by a *HISR* model.The sea anemone can be modeled using a five level *HISR* in which the base, the column, the upper disk, the tentacles, and the mouth are created as root, the first level, and so on. Tentacles are positioned using an implicit surface based phyllotactic technique. Global deformation can be applied to any part of the *HISR* model. The dimension of the anemone and the orientations of the tentacles are easily controlled through global affine transformations. Global bending deformations change the shape of the tentacles and the body.

Since each surface in a *HISR* model is defined locally to its parent hierarchy, it is especially easy to create new models from existing *HISR* model by adding more hierarchies of other objects. Figure 9 (top) shows that the five level sea anemone *HISR* model has been added as a sub-*HISR* of a rock surface. Using an *HPCM* for the rock, the exact contact surface is modeled between the rock and the base of the sea anemone, presenting a vivid scene in which the sea anemone is attached to the environment.

The structure of the *HISR* makes it very easy to create different types of sea anemones using hierarchical local and global operations. In Figure 9 (bottom), different sea anemones are created from the same five-level *HISR* model using different global affine deformations. The anemones are again added as sub-*HISRs* of a rock surface. Precise contact surfaces between the rock and each anemone are created automatically by an *HPCM* hierarchy for the rock. This shows the flexibility and the efficiency of the proposed method in modeling complex objects.

In addition, using *HISR*'s hierarchical local and global deformation operations, the sea anemone's shape deformation can be easily simulated. An animation of the escape response of the sea anemone from a starfish [Nur et al. 2001] shows the dynamic shape control of the *HISR* and its efficiency in animating the *HISR* models.

4.3 The Distributor Caps

HISR is also suitable for building constructive solid geometric models with its *HCSG* hierarchy. Figure 10 shows two distributor cap models built from *HISR* , i.e. a four cylinder distributor cap and a six cylinder distributor cap. Different *HISRs* have been used for these models as shown in Figure 11, illustrating different control ability of the *HISR* model.

Figure 10: Distributor caps built with different *HISR* models

Both distributor caps are modeled using a four level *HISR*. The first level *HDifference* is used to create the hollow inside of the cap. The second level is the cap's base modeled by a disk. In the

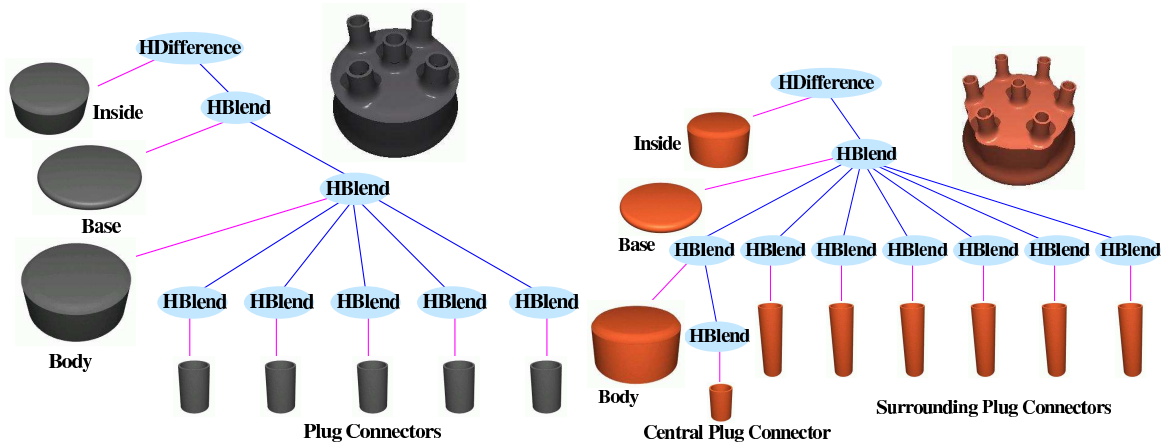

Figure 11: Two different *HISRs* for modeling distributor cap

four cylinder distributor cap model, the body is added as the third level and all plug connectors are added as local details of the cap's body at the fourth level. In the six cylinder distributor cap model, the body and the surrounding plug connectors are added at the third level, while the central plug connector is added as local detail of the body at the fourth level. Both *HISR* models use primitive surface for the base and complex surfaces for the cap's body and plugs as local details.

Figure 12: Globally deformed distributor caps

Figure 12 shows the distorted caps after a global bending operation is applied to the base and the body. As a result, the plugs are automatically adjusted to orient along the new deformed surface. The integrity of the distributor caps are maintained as expected.

5 Conclusions and Future Works

We have presented a new method for implicit modeling, *HISR* - hierarchical implicit surface refinement. Using a hierarchical representation of implicit surfaces, a complex object is modeled as a hierarchy of implicit surfaces in which each node is constrained by its parent. The advantage of the method is that both hierarchical local refinement and global surface deformation can be performed arbitrarily at any level of the surface while maintaining the integrity of the object. In addition, it combines blending, controlled blending, CSG Boolean operations, and precise contact modeling under a general framework, making it suitable for a variety of applications in geometric modeling and animation. The local constraints and the hierarchy itself provide fine control over the composition of different level of surfaces.

Applications in biological modeling, engineering modeling, and interactive modeling show that the new method provides an intuitive, flexible, and efficient way of building complex implicit models, representing deformed objects, and animating the deformations of the surfaces.

Although skeleton based implicit surfaces have been used to represent surfaces in our description, other implicit surface representations, such as functional representation [Pasko et al. 1995], are applicable as well. As mentioned in section 3.1.2, our current method uses the simple splitting scheme and surface based splitting scheme to create local details. A more general technique could be used to find an optimal group of primitives approximating local details. For example, a similar method to Muraki's blobby model [Muraki 1991] could be used for such purpose. An efficient optimization process should be developed for practical use of such a scheme since Muraki's method is very computational expensive. A direct rendering method is also expected to extend the efficiency of the *HISR*. To improve the interaction during local refinement, we are currently working on a faster implicit surface visualization method based on an incremental technique [Galin and Akkouche 1998]. An alternative is to draw only the contour lines for interactive visualization as in [Bremer and Hughes 1998].

In addition, applications of the *HISR* in other areas are well worth exploring, such as volume visualization, detail of surfaces, and etc. We have been working on building an animation environment based on the *HISR* to animate the surface deformations.

6 Acknowledgements

The authors would like to thank the many students, past and present who have contributed toward the implicit modeling research project at the University of Calgary. We would also like to thank the department of computer science and the MACI project for the use of a large cluster of workstations for high performance computing. This work is partially sponsored by the Natural Sciences and Engineering Research Council.

References

BLINN, J. 1982. A generalization of algebraic surface drawing. *ACM Transactions on Graphics 1*, 3, 235–256.

BLOOMENTHAL, J., AND SHOEMAKE, K. 1991. Convolution surfaces. In *Proceedings of SIGGRAPH 91*, ACM Press /

ACM SIGGRAPH, Computer Graphics Proceedings, Annual Conference Series, ACM, 251–256.

BLOOMENTHAL, J. 1997. *Introduction to Implicit Surfaces*. Morgan Kaufmann, ISBN 1-55860-233-X. Edited by Jules Bloomenthal With Chandrajit Bajaj, Jim Blinn, Marie-Paule Cani-Gascuel, Alyn Rockwood, Brian Wyvill, and Geoff Wyvill.

BREMER, D. J., AND HUGHES, J. F. 1998. Rapid approximate silhouette rendering of implicit surfaces. In *The Third International Workshop on Implicit Surface (Proc. Implicit Surface 98)*, Eurographics/SIGGRAPH, 155–164. Seattle, USA.

CANI-GASCUEL, M.-P. 1998. Layered models with implicit surfaces. In *Proceedings of Graphics Interface '98*, K. Booth and A. Fournier, Eds., Graphics Interface, 201–208. ISBN 0-9695338-6-1.

DESBRUN, M., AND CANI-GASCUEL, M.-P. 1998. Active implicit surface for animation. In *Proceedings of Graphics Interface '98*, K. Booth and A. Fournier, Eds., Graphics Interface, 143–150. ISBN 0-9695338-6-1.

DESBRUN, M., TSINGOS, N., AND GASCUEL, M. P. 1996. Adaptive sampling of implicit surfaces for interactive modeling and animation. *Computer Graphics Forum (Proceedings of Eurographics 96) 15* (December), 319–326.

ELLIOT, J. K., ROSS, D. M., PATHIRANA, C., MIAO, S., ANDERSON, R. J., SINGER, P., KOKKE, W. C. M. C., AND AYER, W. A. 1989. Induction of Swimming in Stomphia (Anthozoa: Actiniaria) by Imbricatine, a Metabolite of the Asteroid Dermasterias Imbricata. *Biological Bulletin 176* (April), 73–78.

FORSEY, D. R., AND BARTELS, R. H. 1988. Hierarchical b-spline refinement. In *Proceedings of SIGGRAPH 88*, ACM Press / ACM SIGGRAPH, Computer Graphics Proceedings, Annual Conference Series, ACM, 205–212.

FRISKEN, S. F., PERRY, R. N., ROCKWOOD, A., AND JONES, T. 2000. Adaptively sampled fields: A general representation of shape for computer graphics. In *Proceedings of SIGGRAPH 2000*, ACM Press / ACM SIGGRAPH, Computer Graphics Proceedings, Annual Conference Series, ACM, 249–254.

GALIN, E., AND AKKOUCHE, S. 1998. Incremental techniques for implicit surface modeling. In *Proceedings of Computer Graphics International '98*, IEEE Computer Society Press, IEEE Computer Society, 312–323. Hannover, Germany.

GASCUEL, M.-P. 1993. An implicit formulation for precise contact modeling between flexible solids. In *Proceedings of SIGGRAPH 93*, ACM Press / ACM SIGGRAPH, Computer Graphics Proceedings, Annual Conference Series, ACM, 313–320.

GUY, A., AND WYVILL, B. 1995. Controlled blending for implicit surfaces. In *Implicit Surfaces '95*, Eurographics/SIGGRAPH.

HARBO, R. M. 1949. *Whelks to Whales. Coastal Marine Life of the Pacific Northwest*. Harbour Publishing, ISBN: 1-55017-183-6, BC, Canada.

HART, J. 1996. Sphere tracing: A geometric method for the antialiased ray tracing of implicit surfaces. *The Visual Computer*, 10 (December), 527–545.

KALRA, D., AND BARR, A. 1989. Guaranteed ray intersections with implicit functions. In *Proceedings of SIGGRAPH 89*, ACM Press / ACM SIGGRAPH, Computer Graphics Proceedings, Annual Conference Series, ACM, 297–306.

LIANG, X., AND WYVILL, B. 2001. Hierarchical implicit surface refinement. In *Proceedings of Computer Graphics International 2001*, IEEE Computer Society Press, IEEE Computer Society, 291–298. Hong Kong, China.

MARKOSIAN, L., COHEN, J. M., CRULLI, T., AND HUGHES, J. 1999. Skin: A constructive approach to modeling free-form shapes. In *Proceedings of SIGGRAPH 99*, ACM Press / ACM SIGGRAPH, New York, Computer Graphics Proceedings, Annual Conference Series, ACM, 393–400. Los Angeles, California.

MURAKI, S. 1991. Volumetric shape description of range data using 'blobby model'. In *Proceedings of SIGGRAPH 91*, ACM Press / ACM SIGGRAPH, Computer Graphics Proceedings, Annual Conference Series, ACM, 227–235.

NUR, M. A., LIANG, X., WYVILL, B., AND BOURNE, G. B. 2001. Animating the escape response of *stomphia coccinea* from *dermasterias imbricata* modeled using implicit surfaces. In *Proceedings of Computer Graphics International 2001*, IEEE Computer Society Press, IEEE Computer Society, 81–88. Hong Kong, China.

PASKO, A., ADZHIEV, V., SOURIN, A., AND SAVCHENKO, V. 1995. Function representation in geometric modeling: concepts, implementation and applications. *The Visual Computer 2*, 8, 429–446.

RICCI. 1973. Constructive geometry for computer graphics. *computer journal 16*, 2 (May), 157–160.

SNYDER, J. M. 1992. Interval analysis for computer graphics. In *Proceedings of SIGGRAPH 92*, ACM Press / ACM SIGGRAPH, Computer Graphics Proceedings, Annual Conference Series, ACM, 122–130.

STANDER, B., AND HART, J. 1998. Guaranteeing the topology of an implicit surface polygonization. In *Proceedings of SIGGRAPH 98*, ACM Press / ACM SIGGRAPH, Computer Graphics Proceedings, Annual Conference Series, ACM, 69–76.

STEPHENSON, T. A. 1934. *The British Sea Anemones*. Arlard and Son, Limited, London.

WYVILL, G., AND TROTMAN, A. 1990. Ray tracing soft objects. In *Proc. CG International 90*, Springer Verlag, IEEE Computer Society, 469–476.

WYVILL, B., MCPHEETERS, C., AND WYVILL, G. 1986. Animating Soft Objects. *The Visual Computer 2*, 4 (February), 235–242.

WYVILL, G., MCPHEETERS, C., AND WYVILL, B. 1986. Data Structure for Soft Objects. *The Visual Computer 2*, 4 (February), 227–234.

WYVILL, B., GALIN, E., AND GUY, A. 1999. Extending The CSG Tree. Warping, Blending and Boolean Operations in an Implicit Surface Modeling System. *Computer Graphics Forum 18*, 2 (June), 149–158.