

# Deformable Solid Modeling via Medial Sampling and Displacement Subdivision

by  
Andrew Lewis Thall

A dissertation submitted to the faculty of The University of North Carolina at Chapel Hill in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Computer Science.

Chapel Hill  
2004

Approved by:

---

Stephen M. Pizer, Advisor

---

Gary Bishop, Reader

---

Dinesh Manocha, Reader

---

Daniel Fritsch, Reader

---

Sarang Joshi, Reader

Copyright © 2004  
Andrew Lewis Thall  
All rights reserved

**ANDREW LEWIS THALL. Deformable Solid Modeling via Medial  
Sampling and Displacement Subdivision.  
(Under the direction of Stephen M. Pizer.)**

**ABSTRACT**

*Discrete m-reps* use tolerance-based, sampled medial skeleta as an underlying framework for boundaries defined by displaced subdivision surfaces. They provide local and global shape information, combining the strengths of multiscale skeletal modeling with the multi-resolution, deformation and shading properties afforded by subdivision surfaces. Hierarchically linked medial figures allow figural, object-based deformation, and their stability of object representation with respect to boundary perturbation shows advantages of tolerance-based medial representations over Blum axis and Voronoi-based skeletal models.

M-rep models provide new approaches to traditional computer graphics modeling, to physically based modeling and simulation, and to image-analysis, segmentation and display, by combining local and object-level deformability and by explicitly including object-scale, tolerance and hierarchical level-of-detail. Sampled medial representations combine the solid modeling capabilities of constructive solid geometry with the flexibility of traditional b-reps, to which they add multiscale medial and boundary deformations parameterized by an object-based coordinate system.

This thesis research encompassed conceptual development on discrete m-reps and their implementation for MIDAG (the Medical Image Display and Analysis Group) at UNC-Chapel Hill. Prototype and application code was created to support the following: medial atoms in 3D that included a quaternion frame to establish a local coordinate system; data structures for medial mesh topologies; a new algorithm for interpolating Catmull-Clark subdivision surfaces for m-rep boundaries; a medially based coordinate system parameterizing the m-rep boundary, interior, and local exterior; displacement texturing and displacement meshing of m-rep boundaries; methods of medially based deformation; figure/subfigure blending by implicit surface methods or (with Qiong Han) using remeshing of subdivision boundaries; and C++ code libraries for m-rep modeling in geometric design and image-segmentation applications.

Along with discussion of these achievements, this document also includes discussions of current m-rep applications and of design-methodology issues for m-rep-based modeling systems.

For Nancy and Robby.

**On writing**

Of the making of many books there is no end;  
and much study is a weariness of the flesh.

—Ecclesiastes 12:12

**On the thesis topic**

We asked our captain what course of action he proposed against  
so formidable a beast. He thought judiciously for a moment  
and then replied, “I think I shall praise it.”

—epigram on a book of love poems, *Praise*, by Robert Hass

**On setting forth**

“In the Great House and the House of Fire,  
on the Day when all days and years are numbered,  
let my name be given back to me.”

—the Osiris-scribe Ani, ca. 1500 B.C.E.

# Acknowledgments

In a project spanning so many years, there are countless people to whom thanks are due. First and foremost, to Steve Pizer—the work would never have been done without his insights, his goading, his advice and friendship; to my other committee members, as well as to Lee Westover, who sat on my committee in the early years; to Turner Whitted, for ideas and advice in the early days of the work; to Lee Nackman, for the inspiration of his early work and the pleasure of a conversation in 2001; To everyone at MIDAG who worked with me on medial matters, especially the inestimably helpful Graham Gash; to Tom Fletcher and Paul Yushkevich and Jessica Crouch, my comrades-in-arms, for lots of ideas and discussion along the way. to Qiong Han, my collaborator on the m-rep figure-subfigure blending work; to Jim Damon, a superb teacher and colleague; to Eitan Grinspun at Caltech for his advice on proximity tests on subdivision surfaces; to Rob Katz, as my spiritual advisor, and to Janet Jones for her excellent friendship and for helping me to survive my too numerous run-ins with University Administration; to Matt Lavoie, old friend, for giving me a place to crash when I was in town to finish off the writing; to all my friends in the department, whether former office-mates, students or faculty. to Dr. Robert Cupper, my department chairman and mentor at Allegheny College, for his faith that my Ph.D. was just a matter of time and for convincing the Allegheny administration of that fact.

Special thanks to John, Paul, George and Ringo.

This work received support from National Cancer Institute grant PO1-CA47982. It was directly funded by National Science Foundation SGER grant (Special Grant for Exploratory Research) CCR-9910419 for 1999-2000—special thanks to Tim Quigg for help negotiating that grant. I also received a Link Foundation fellowship for the year 1998-1999 to support M-rep research.

To Robert McCabe, in fond memory of those discussions of Fourier transformations; to Joseph Thall, in love and respect. To my father and mother, Don and Liz Thall. And to Nancy and Robby, again and always, for all the time I spent at home when I should have been in my office working.

# Contents

<b>Acknowledgments</b>	<b>vi</b>
<b>List of Tables</b>	<b>xii</b>
<b>List of Figures</b>	<b>xiii</b>
<b>Glossary</b>	<b>xvi</b>
<b>1 Overview</b>	<b>1</b>
1.1 Conventional geometric modeling primitives . . . . .	2
1.1.1 Boundary representations . . . . .	2
1.1.2 Solid modeling primitives . . . . .	4
1.1.3 Volumetric models . . . . .	5
1.1.4 Medial modeling primitives . . . . .	6
1.1.5 Image-based rendering techniques . . . . .	7
1.2 Thesis statement, claims and contributions . . . . .	7
1.3 Advantages of m-reps . . . . .	11
1.3.1 Of medial primitives . . . . .	11
1.3.2 Of tolerance-based methods . . . . .	12
1.3.3 Of hierarchical object definition . . . . .	13
1.3.4 Of displaced subdivision surfaces with sampled medial skeleta	14
1.3.5 Of primitives designed as robust statistical shape descriptors .	14
1.4 Dissertation outline . . . . .	15
<b>2 Related Work</b>	<b>17</b>
2.1 Medial skeleta: the Blum symmetric axis and Voronoi-based techniques	18
2.1.1 Geometric analysis—b-rep to m-rep versus m-rep to b-rep . .	18
2.1.2 Medial-primitive representations in CG . . . . .	20
2.2 Implicit surfaces, volume modeling and convolution surfaces . . . . .	23

2.3	Multiscale and tolerance-based medial methods . . . . .	24
2.3.1	Object description by scale-based representations . . . . .	25
2.3.2	Multiscale methods in computer graphics . . . . .	28
2.4	Other approaches to M-reps . . . . .	28
2.4.1	Volume-rendering using MMAs . . . . .	29
2.4.2	Core atoms as fuzzy shape descriptors . . . . .	29
2.4.3	Implicit boundary m-reps . . . . .	30
2.4.4	Continuous medial surface methods . . . . .	30
2.5	Subdivision surfaces . . . . .	32
2.5.1	Interpolating and approximating surfaces . . . . .	32
2.5.2	Parameterization . . . . .	33
2.5.3	Displacement subdivision surfaces . . . . .	33
2.5.4	Boolean operations on subdivision surfaces . . . . .	34
2.5.5	Physically based operations on subdivision surfaces . . . . .	35
2.6	Concluding remarks on prior work . . . . .	35
<b>3</b>	<b>Object Description by Sampled Medial Sheets</b>	<b>36</b>
3.1	The sampled medial sheet . . . . .	37
3.1.1	The medial atom . . . . .	38
3.1.2	The single-figure medial mesh . . . . .	40
3.1.3	Invariance under similarity transformation . . . . .	43
3.1.4	Connectivity and topology of sampled medial meshes . . . . .	44
3.1.5	Medial coordinates and correspondence for quadmeshes . . . . .	47
3.1.6	Restrictions on boundary curvature of m-reps . . . . .	50
3.2	Figural hierarchies . . . . .	51
3.2.1	Representational polymorphism . . . . .	56
3.3	Interpolation of medial atoms . . . . .	56
3.3.1	Interpolating a continuous Blum axis from a sampled mesh . . . . .	58
3.3.2	Interpolating a Blum axis from an m-rep-based b-rep . . . . .	61
3.3.3	Interpolating a multiscale axis from $(u, v, t)$ correspondence . . . . .	62
3.4	Topology-preserving deformation . . . . .	65
3.4.1	Medially based object deformations as a primitive operations . . . . .	65
3.4.2	Boundary deformation . . . . .	68
3.5	Drawbacks and limitations of m-reps . . . . .	69
3.5.1	Quadmeshes and their disadvantages . . . . .	69

3.5.2	Corners, creases, and other surface discontinuities . . . . .	70
3.5.3	The Correspondence Problem . . . . .	70
3.5.4	Discrete m-reps vs. cm-reps . . . . .	70
3.6	Concluding remarks on sampled medial sheets . . . . .	71
<b>4</b>	<b>M-rep Boundary Fitting by Interpolating Subdivision Surfaces</b>	<b>72</b>
4.1	Iteratively interpolating subdivision surfaces . . . . .	73
4.1.1	Interpolating Catmull-Clark boundaries . . . . .	75
4.1.2	IIS-surfaces—regular vertices . . . . .	80
4.1.3	IIS-surfaces—irregular vertex limit positions and inverses . . . .	84
4.1.4	Error metrics for IIS-surfaces relative to medially implied boundaries . . . . .	87
4.1.5	Accurate normal interpolation by modified IIS-surfaces . . . . .	91
4.1.6	Interpolation of other boundary attributes . . . . .	92
4.1.7	Drawbacks and limitations of IIS-surfaces . . . . .	95
4.2	Proximity tests on m-rep boundaries . . . . .	96
4.2.1	The Phong-normal nearpoint estimate . . . . .	97
4.2.2	Drawbacks, extensions and improvements . . . . .	100
4.3	Interpolating the medial structure of subdivision solids . . . . .	101
4.3.1	Algorithm and implementation . . . . .	102
4.3.2	Discussion of Phong-normal medial axis . . . . .	103
4.4	Concluding remarks on medially implied subdivision boundaries . . . .	104
<b>5</b>	<b>Object Modeling using M-rep Subdivision Solids</b>	<b>105</b>
5.1	Subdivision Boundary Displacement: Texture and Mesh Approaches . . . .	106
5.1.1	Image and displacement textures for m-rep boundaries . . . . .	106
5.1.2	Displacement meshes for m-rep subdivision boundaries . . . . .	110
5.1.3	Modeling using tolerance-based boundary displacement . . . . .	113
5.2	Multifigure Modeling of M-reps (co-author: Qiong Han) . . . . .	115
5.2.1	Medial coordinates for subfigure positioning . . . . .	116
5.2.2	Smoothly joining figure and subfigure by remeshing . . . . .	120
5.2.3	Joining figure and subfigure by remeshing: implementation . . . .	121
5.2.4	Fitting a polygonal model with a blended, multifigure m-rep . . . .	125
5.2.5	Implementation issues and drawbacks . . . . .	128
5.3	Concluding remarks . . . . .	134

<b>6</b>	<b>Discrete M-Rep Modeling Systems</b>	<b>136</b>
6.1	Rakshasa: m-reps for computer graphics and CAGD . . . . .	136
6.1.1	Medial atoms and medial mesh topologies . . . . .	138
6.1.2	Model construction and medially based deformation . . . . .	138
6.1.3	Medial and boundary interpolation . . . . .	139
6.1.4	Catmull-Clark subdivision boundaries . . . . .	140
6.1.5	Image and displacement textures . . . . .	140
6.1.6	Drawbacks of Rakshasa . . . . .	140
6.2	Seurat and Pablo: m-reps for modeling and image-segmentation . . .	142
6.2.1	Pablo functionality . . . . .	143
6.2.2	Seurat functionality . . . . .	147
6.2.3	Drawbacks in Pablo and Seurat . . . . .	151
6.3	M-rep research applications . . . . .	152
6.3.1	Current Pablo research . . . . .	153
6.3.2	Styner: m-reps for statistical shape analysis . . . . .	155
6.3.3	Crouch: finite element modeling using m-reps . . . . .	155
6.3.4	Fletcher: Lie algebras and principal geodesic analysis . . . . .	156
6.3.5	Application of Fletcher’s methods for m-rep animation . . . . .	157
6.4	Concluding remarks on modeling systems . . . . .	158
<b>7</b>	<b>Discussion and Future Work</b>	<b>160</b>
7.1	Tools and paradigms for multiscale, multigure modeling . . . . .	161
7.1.1	Medial and tolerance-based object deformation . . . . .	162
7.1.2	Multigure DAGs and tree hierarchies for pseudo-CSG . . . . .	166
7.1.3	Blobby modeling and virtual clay modeling paradigms . . . . .	166
7.1.4	Precision modeling paradigms . . . . .	167
7.1.5	Last remarks on m-rep modeling methodology . . . . .	168
7.2	Improvements to current m-rep technology . . . . .	169
7.2.1	Better m-rep boundary surfaces . . . . .	169
7.2.2	Improvements in boundary-parameter interpolation . . . . .	169
7.2.3	More systematic boundary and internal displacements . . . . .	170
7.2.4	Improved interfaces for multigure design . . . . .	171
7.2.5	Trimeshed boundaries and new parameterizations . . . . .	172
7.2.6	Trimeshed medial axes and interactive resampling . . . . .	173
7.2.7	Fitting m-reps to b-reps or objects in volumetric data . . . . .	174

7.2.8	Improvements in medial techniques . . . . .	175
7.2.9	Improvements in corner and edge specification . . . . .	175
7.3	Directions for future m-rep applications . . . . .	176
7.3.1	M-reps in medical image analysis . . . . .	176
7.3.2	M-reps in computer graphics and CAD . . . . .	179
7.3.3	Final remarks on future m-rep applications . . . . .	184
7.4	Personal words . . . . .	185
	<b>Bibliography</b>	<b>186</b>

# List of Tables

1.1	Modeling primitives and their features. . . . .	12
-----	---	----

# List of Figures

1.1	M-rep based models . . . . .	1
1.2	Example medial atom and medial end-atom . . . . .	9
2.1	The Blum medial axis vs. the MMA. . . . .	27
2.2	M-rep figure/subfigure blending using an implicit surface . . . . .	30
2.3	A 2D continuous m-rep ( <i>cm-rep</i> ) of a vertebra in cross-section . . . . .	31
3.1	A medial atom and frame relative to the medial tangent space $T_p(\mathcal{M})$ . . . . .	39
3.2	Width-proportional sampling of fat vs. thin object . . . . .	40
3.3	Two m-reps allowing varying medial deformation . . . . .	41
3.4	Catastrophic behavior of $\vec{v}_1$ and $\vec{v}_2$ at edge of medial sheet . . . . .	42
3.5	Examples of medial mesh topologies (1) . . . . .	45
3.6	Examples of medial mesh topologies (2) . . . . .	46
3.7	Boundaries with a <i>non-Blum</i> medial axis . . . . .	48
3.8	Parametric correspondence of medial atoms and boundary involutes . . . . .	49
3.9	An m-rep with a bend disproportionate to object width . . . . .	51
3.10	A continuous m-rep with a knotted boundary . . . . .	52
3.11	Inter- and intra-figural relationships . . . . .	53
3.12	An illustration of non-branching of MMAs in scale-space . . . . .	54
3.13	A 2D m-rep with attached subfigures . . . . .	55
3.14	A 3D m-rep with figure and subfigure generated by Pablo . . . . .	55
3.15	A 3D object with a hole, achieved by a 1- or 2-figure m-rep . . . . .	57
3.16	2D m-reps that fail to have “Blumness” . . . . .	59
3.17	Interpolating a medial atom given a boundary correspondence . . . . .	64
3.18	Interpolating a medial axis using boundary correspondence . . . . .	64
3.19	Deforming an object by medial mesh modification (1) . . . . .	66
3.20	Deforming an object by medial mesh modification (2) . . . . .	67
4.1	(Tom Fletcher) A figure-subfigure blend based on an implicit-function . . . . .	74

4.2	Medial mesh with subdivision boundaries . . . . .	76
4.3	<i>Near</i> -interpolation of medial involutes by subdivision boundary. . . . .	76
4.4	An initial polygonal model for subdivision tests . . . . .	77
4.5	One level of interpolating subdivision applied to polygonal model . . . . .	78
4.6	One level of interpolating subdivision applied to polygonal model . . . . .	79
4.7	Vertex labeling in a regular 1-neighborhood . . . . .	81
4.8	Regularizing the neighborhood of an irregular vertex . . . . .	87
4.9	Iterative subdivision applied to a coarse torus-mesh . . . . .	88
4.10	Iterative subdivision applied to a pyramid-mesh . . . . .	89
4.11	An m-rep boundary generated by normal-interpolating IIS-surfaces . . . . .	93
4.12	A bone modeled by ordinary and normal-interpolating IIS-surfaces . . . . .	94
4.13	Endcap interpolation using IIS-surfaces . . . . .	96
4.14	Euclidean minimum distance vs. <i>Phong-normal</i> projection nearpoint . . . . .	97
4.15	A nearpoint defined by normal projection may actually be a farpoint . . . . .	98
4.16	Finding a Phong-normal intepolant intersecting a position $\mathbf{x}$ . . . . .	99
4.17	Finding an interpolating medial atom using bisection search . . . . .	102
4.18	An m-rep resampled by Phong-normal medial interpolation . . . . .	103
5.1	A redwood image-texture and associated displacement map . . . . .	108
5.2	Image-based displacement mapping of an m-rep slab figure . . . . .	109
5.3	Edge view of displaced m-rep boundary . . . . .	110
5.4	M-rep textured with displacement meshes . . . . .	112
5.5	An indentation figure with visible hinge atoms . . . . .	117
5.6	Placement of a subfigure in $(u, v, t)$ parameter space of parent figure . . . . .	119
5.7	Examples of protrusion and indentation subfigures . . . . .	122
5.8	Figure-subfigure intersection curves . . . . .	123
5.9	Dilation of intersection curves on parent-figures . . . . .	124
5.10	Detail of figure-subfigure remeshing in region of join . . . . .	126
5.11	Adjustment of blending parameters for protrusion subfigure . . . . .	127
5.12	Adjustment of blending parameters for indentation subfigure . . . . .	127
5.13	The <code>big_spider.ply</code> model . . . . .	128
5.14	Medial structure of the spider m-rep . . . . .	129
5.15	Subdivision boundaries for the spider m-rep (1) . . . . .	130
5.16	Subdivision boundaries for the spider m-rep (2) . . . . .	131
5.17	Close-up details of spider m-rep figure-subfigure blends (1) . . . . .	132
5.18	Close-up details of spider m-rep figure-subfigure blends (2) . . . . .	133

5.19	Positioning of subfigures across problematic crest regions . . . . .	134
6.1	Curling and twisting operations on m-reps . . . . .	139
6.2	Image and displacement mapping in Rakshasa . . . . .	141
6.3	Prototype multiframe m-reps produced by SCAMP . . . . .	142
6.4	Multiframe brain-ventricle m-rep modeled in Pablo . . . . .	143
6.5	Pablo kidney m-rep with subdivision boundaries . . . . .	144
6.6	Pablo kidney m-rep in image data with cut-planes . . . . .	146
6.7	The three keyframe m-reps from the spider animation . . . . .	159

# Glossary

Throughout this dissertation, special terminology is used to convey precise meanings or to draw specific distinctions. This glossary is provided as a convenient reference for the reader.

**Displacement Subdivision Surface** Any of a class of subdivision-based surfaces that include a boundary displacement field applied to the vertices at any given subdivision level. The field may be stored on a *per vertex* basis in the subdivision meshes or as values in displacement maps.

**Iteratively Interpolating Subdivision Surface** (*IIS-surface*) An interpolating subdivision surface, created by iteratively solving for an initializing mesh that will interpolate the desired mesh locations.

**Medial involute** *Involute* has several geometric meanings; in the context of this paper, the *medial involutes* (typically paired) are opposing points on the boundary of a solid equidistant from their associated point on the medial axis. In terms of the Blum axis, they are the boundary points tangential to the maximal enclosed circle/sphere with the medial point at its center.

**M-rep** A *medial representation* for a solid object, basing the shape geometry on medial positions and a medial radius function. A *discrete m-rep* is a medial representation based on meshes of discrete medial atoms; a *cm-rep* is a continuous, spline-based medial representation. In this dissertation, m-rep, unless qualified, will typically refer to a discrete m-rep, and the unabbreviated *medial representation* will be used for the more general definition.

**Modeling** A word with too many diverse meanings. In this document, *modeling* will always be used for the process of *shape synthesis* to create a 3D geometric model for use in a computer graphics or image analysis application. *Model*, likewise, will always be used (unless otherwise qualified) for the product of such a shape

synthesis. This, as distinguished from a *statistical* model, especially statistical shape descriptors or statistical variations placed on a 3D geometric model for shape analysis or procedural instancing.

**Multiresolution vs. multiscale** *Multiresolution* methods utilize level-of-detail in object definition and encompass spherical harmonics, wavelet-based methods, and multilevel editing of subdivision surfaces. *Multiscale* methods are distinguished by their use of multiple measurement apertures—such as those afforded by Gaussian scale spaces or width-based sampling tolerances—in place of infinitesimally precise definition of geometry based on Euclidean coordinates.

**Pablo and Rakshasa** *Pablo* is a 3D model-building and image-analysis tool which uses m-reps for model-based image segmentation. Pablo was developed by a team of researchers at MIDAG. *Rakshasa* is a similar modeling tool developed by me for discrete m-reps with emphasis on subdivision boundaries with displacement maps and on medial-based deformation.

**Regular quadrilateral mesh** A boundary mesh having only quadrilateral faces. A regular vertex is one with only quadrilateral faces in its 1-neighborhood.

**Representational polymorphism** The ability to represent a shape by different descriptions; e.g., m-reps and CSG provide this polymorphism; Blum medial descriptions do not.

**Representational transparency** A quality of shape design software, whereby models are created by shape-specifying operators and the underlying data representation is invisible to the user.

# Chapter 1

## Overview

In the little more than 40 years since computer-based 3D geometric modeling has been practiced, the choice of geometric primitive has controlled the capabilities in terms of both modeling and rendering. Applications in computer-aided design (*CAD*), commercial advertising, motion picture special effects, physically based simulation and dynamics all have been algorithmically dependent on their respective choices of carriers for the geometric information their models require. Trends in graphics hardware development have tended toward simplifications of geometry allowing highly parallel and pipelined acceleration of rendering, with the use of image-textures, bump and displacement maps, and even procedural textures making up for the lack of underlying geometric complexity. Image-based rendering (*IBR*) is another example, where all geometry is reduced to view-specific height-field information to give  $\mathcal{O}(1)$ -time rendering, independent of scene complexity. For modeling purposes, geometric primitives have been primarily surface-based—polygons, splines, or subdivision surfaces—or, in CAD applications, solid models based on constructive solid geometry (*CSG*) performed on simple geometric solid primitives.

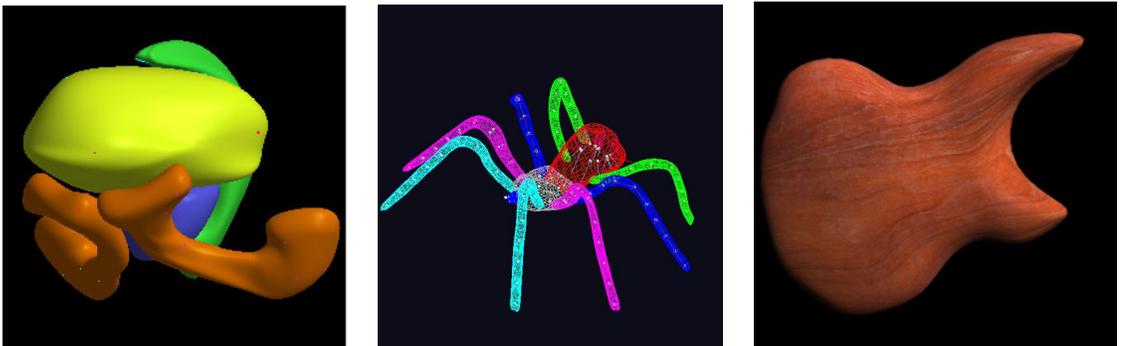


Figure 1.1: M-rep based models.

This research introduces the *discrete m-rep*<sup>1</sup>, a versatile solid-modeling primitive based on sampled, tolerance-based medial axes with small-scale boundary displacements. M-reps are intended to deliver greater capabilities in computer graphics modeling and rendering, in image analysis (as a superior 3D shape descriptor for segmentation and registration tasks), and in physically based simulation. The work shows ways in which m-reps can work within the standard modeling and rendering paradigms and explores their strengths and weaknesses *vis-a-vis* traditional techniques. Original contributions are also made in interpolating subdivision surfaces for object boundaries based on m-reps, finding distance and near-point functions for such objects, and constructing Blum medial axes for them.

This chapter provides a brief overview of conventional modeling primitives and explains their common drawback—*overprecision* or *intolerance*. Such overprecision, as will be explained below, provides too much information at the wrong scale and at the wrong stages during shape analysis, shape synthesis, or shape display, rather than providing appropriate shape information at each given stage in the respective process. Following the overview, I propose the central thesis of this work, establishing its claims and listing completed contributions and accomplishments. The chapter then discusses ways in which m-reps overcome drawbacks of conventional primitives, and it concludes with a chapter outline of the dissertation.

## 1.1 Conventional geometric modeling primitives

Geometric modeling and rendering primitives can be grouped into a number of broad categories (with a good deal of overlap), the most important of which are b-reps, solid models, volumetric representations, medial models, and IBR models.

### 1.1.1 Boundary representations

The simplest b-reps are first-order surfaces, i.e., triangulated or polygonalized tiles, joined edgewise to produce polyhedra or approximations to smooth surfaces. These have a long history in CG and modeling and are typical final-output primitives for rendering APIs such as OpenGL or DirectX. There are well-established methods for interpolating lighting effects, normals, and image or displacement textures across polygonalized surfaces. **Main drawbacks:** (a) they are fixed-scale (rather than

---

<sup>1</sup>from *medial representation*, akin to b-rep, *boundary representation*

multiresolution) primitives and require considerable work for model simplification; (b) they can only approximate smooth surfaces, requiring properties such as surface normals and boundary curvature to be either estimated or computed by assumptions about the object geometry not based on the surface tiling.

Higher order b-reps can be based on spline patches or subdivision. In modeling, these can interpolate known boundary positions and normals or be used for free-form surface design. Typically, spline-based surfaces are subdivided and polygonalized for rendering and thus can be rendered to any given image-scale/resolution. They have well-defined normals and curvatures within each patch. **Main drawbacks:** (a) it is difficult to maintain high-order surface continuity between adjacent patches on a surface, especially when tiling objects of arbitrary topology; (b) patch-based representations are inherently *local* and give little explicit information about solid shape properties..

The development of subdivision surfaces in the 1970's and their rise in popularity in the 1990's was in response to the above drawbacks; they established their usefulness as a b-rep for representing surfaces of solids with arbitrary Euler number (i.e., with holes) and in providing for intuitive, multilevel mesh editing. Their ability to either interpolate or approximate boundary positions made them ideal for capturing surface geometry from 3D scanning devices, and techniques were quickly developed for mapping texture coordinates to them and for creating ridges, sharp edges and cusps. Subdivision surfaces are generalizations of spline-based ones. Their parametric evaluation and curvature properties near extraordinary points has been studied by Stam [137] and others. In recent years, multiresolution surfaces, based on displaced subdivision surfaces and wavelet methods, have been gaining prominence for their ability to define their surfaces to the necessary, task-related scale. **Main drawback:** subdivision surfaces can be used to bound solid volumes, but their modes of deformation remain based on manipulation of mesh points describing boundary locations; as such, they fail to provide a parametric basis for solid object deformation. This drawback is common to all b-rep primitives, and motivates the use of skeletal methods for manipulating boundary meshes. This will be discussed more fully in Sec. 1.3.

A summary of problems with all common b-reps:

- there is no global (solid) object—only surfaces described by linear or higher order patches (subdivision-bounded solids are an exception to this, as noted above);

- there is no inherent deformability based on natural modes of variation for a particular shape; deformations typically manipulate vertices or control points, or apply axial or free-form deformations (*FFDs*) on the embedding space as per Barr [7] or Sederberg [130], respectively; alternatively, finite-element methods provide physically-based solid deformation, but at a high computational cost;
- critically, there is no notion of object scale or tolerance—points, lines and surfaces are defined at infinitesimal scale, to floating point precision, regardless of the accuracy required—and there is no statistical foundation for describing natural object variability; multiresolution surfaces provide a partial solution to this problem; other insights may be drawn from multiscale, Gaussian scale-space techniques (see Sec. 2.3);
- b-reps provide no inherent figural basis for shape organization.

### 1.1.2 Solid modeling primitives

The use of CSG in CAD applications was the earliest and is still the main practical application of solid modeling primitives. By describing objects by Boolean combinations of simple solid primitives, CSG methods provide global information about object shape and structure, allowing precision design and computation of mechanical properties. **Main drawbacks:** for our purposes, (a) CSG models are not inherently deformable—they’re very *solid* solid models; (b) CSG models are unstable with respect to changes in the modeling primitives used for the Boolean construction—small changes to an additive or subtractive element can have unpredictable effects on the geometry of the final solid; (c) CSG models have no built-in notions of scale and tolerance—objects are defined geometrically in Euclidean space, to either fixed-point or floating-point accuracy. Attempts to do CSG on multiresolution primitives such as objects bounded by subdivision surfaces (e.g., Biermann, et al. [8]) are still in early stages of research.

In the early 1980s, *blobby* modeling and similar implicit techniques were developed by Blinn [11], the Wyvill’s [160, 161], Bloomenthal [15] and others. These provide deformable solid models, bounded by the level-sets of various implicit functions—Gaussian blobs and the like—which can be combined and blended through simple compositing operators on their defining functions. Convolution surfaces, as developed by Bloomenthal and Shoemake [14] and with continuing work by Sherstyuk [132, 133, 134], give another approach to implicit modeling by using convolutions over polygonal

skeletal elements. **Main drawbacks:** (a) as above, there is no inherent notion of scale or tolerance—while it is possible to sample the implicit functions more coarsely, this may introduce artifacts and can’t be done in an object-based fashion because the objects are *defined* by the results of the implicit function evaluation when sampled; (b) also as above, changes to modeling elements can have unpredictable effects on the output, including unintended changes in object topology; (c) blending of objects at joins is automatic, based on the blending of the implicit functions, but it is tricky to impose restrictions on joins, making them sharper or more gradual as desired. While convolution surfaces provide more control of object-level deformation, due to their skeletal structure, fine control is still an issue; (d) in general, implicit models have the opposite problem from b-reps—local, fine-scale deformation (and surface parameterization) is difficult, while global deformation is simpler.

A recent idea in solid modeling is the solid subdivision model of McDonnell et al. [103], building from methods developed for free-form deformation (*FFD*) by MacCracken and Joy [100]. The authors combine subdivision solids—as generalized from tricubic b-spline-based FFD volumes—with physically based models to create *dynamic subdivision solids*, for use in real-time dynamic sculpture with haptic interfaces, so called “virtual clay.” This method will be discussed in more detail in Ch. 2; it is near in spirit to my own goals for a new modeling primitive. **Main drawbacks:** (a) as always, the lack of a multiscale, tolerance-based framework for modeling; (b) no clear framework for multifigural representation.

### 1.1.3 Volumetric models

Volume-rendering techniques such as ray-casting and splatting arose for explicitly displaying 3D datasets, primarily for medical imaging applications. Volume graphics has developed subsequently as a way to render objects by a 3D rasterization to produce a dataset, then volume-rendering for a particular viewpoint; representative research in this area is by Kaufmann [90], Avila [4], and Sclaroff [129]. Volume-primitives have been proposed—e.g., the SGI *OpenGL Volumizer* system [75]—to provide a way to define volume-based “objects” that can be manipulated and deformed. Another approach, hypertextures, imposes a function (frequently, a time-varying, scale-dependent function based on Perlin noise) on a volume of space, which is then evaluated by ray-casting from the eyepoint through the image-plane into the scene, until maximum opacity is achieved.

Besides ray-casting and splat-based methods, implicit surface techniques can al-

ternatively be applied to volume data, treating it as a discretely sampled function and finding level-set surfaces between object interiors and exteriors.

**Main drawbacks:** (a) volumetric techniques allow rendering to image-scale, but they have an “inner scale” fixed by the voxel size—they are not a *tolerant* modeling primitive (see Sec. 1.3.2); (b) surface-definition and rendering are typically slow, though there has been work on hardware speedups; (c) ray-casting and splatting both require recomputation for changes in the viewpoint—tiled level-sets can, however, be rendered as easily as other polygonal primitives; (d) “objects” are generally defined only implicitly, as level-sets or regions of opacity, making manipulation and deformation difficult (OpenGL Volumizer was created to overcome this); and (e) 3D rasterization remains an awkward and slow way to render traditional b-reps.

### 1.1.4 Medial modeling primitives

Recent years have seen a rise in medial-based primitives in both modeling and rendering. Implicit surfaces, in particular the convolution surfaces of Bloomenthal and Sherstyuk, can make use of medial/skeletal elements for object creation and deformation. Skeletal modeling and techniques for *skinning* polyhedral skeleta are finding wider application in computer graphics and animation. This work will be discussed in more detail in Ch. 2, but worth noting here are these researchers: (a) Gascuel, Verroust and Puech [67], who developed a system for animation and collision detection of models based on rigid articulated skeleta fleshed by spline-based deformable boundary surfaces; (b) Markosian et al. [102], who created *Skins* using subdivisions surfaces offset from skeletal elements by the blending of implicit distance functions; (c) Storti et al. [142] in 1997 and Blanding et al. [10] in 1999, who each developed solid editors based on the Blum medial axis [16, 18] and using Voronoi methods for converting between b-rep and skeletal representations; (d) Gagvani [65], who developed techniques for volume animation using skeletal trees for volume rendering and volume graphics.

**Main drawbacks:** (a) many of the techniques use piecewise linear or polyhedral skeletal elements, lacking the generality of a medial primitive based more intimately on the solid geometry; (b) while some of the methods discuss the level-of-detail (*LOD*) editing afforded by pruning a Blum-based medial axis, none of them make the jump to a tolerance-based realization based on *multiscale* medial techniques as developed in the image analysis community. A more detailed critique of current art in medial modeling will be given in the next chapter.

### 1.1.5 Image-based rendering techniques

Along with IBR methods I include methods such as height fields, light-field-rendering, and relief-textures. Pure IBR is a scene-rendering technique with no concept of object—the “modeling primitive” is an entire image or collection of images. IBR greatly enhances scene complexity by replacing geometric models by images acquired either from the real world or from CG rendering. **Main drawback:** the inner scale, pixel-scale, is both fixed and ill-defined, as each viewpoint-dependent pixel samples over a potentially infinite volume of space. This is deliberate to give a constant rendering time, but primitives may not be edited in any simple fashion, as there is no concept of “object” apart from the images themselves. Hybrid methods, like the relief textures of Oliveira [111], combine image-based “objects” in traditional CG scenes; while allowing object-based scene creation, they still have the drawbacks of fixed sampling scale and lack of intrinsic editing and object-level deformability.

## 1.2 Thesis statement, claims and contributions

To confront the drawbacks seen in the above techniques, a modeling primitive should have the following properties:

- precision based on necessary tolerance, with a notion of intrinsic scale and *modeling aperture* (see Sec. 1.3.2),
- medial attributes to give better reflection of global geometry than boundary primitives,
- boundary attributes to give better reflection of fine-scale, local geometry than medial primitives,
- object-based deformability,
- hierarchical organization based on object-shape,
- a framework for incorporating statistical shape variations into a model.

Although many of the primitives discussed above have some subset of these attributes, none of the primitives have the entire set. In particular, the need for tolerance in shape-description, while sometimes understood at a conceptual level, has only rarely

been seen in actual modeling systems (see below in Sec. 1.3.2). The rise of multiresolution methods in surface geometry shows that shape descriptors can effectively encapsulate multiscale information for efficient use in rendering, sidestepping the computational costs otherwise associated with model simplification. The usefulness of medial attributes is acknowledged by the prevalence of skeletal techniques in computer animation, and, similarly, statistical shape variation has been used in CG films to produce populations of similar but distinct objects.

Seeking a modeling primitive combining all of these attributes and driven by the need in *MIDAG*<sup>2</sup> at UNC-Chapel Hill for a modeling system based on multiscale medial methodology, we have developed the m-rep, a geometric representation based on a sampled medial sheet that implies a subdivision-surface boundary. From this work comes the statement of my thesis:

*Discrete m-reps* use tolerance-based, sampled medial skeleta as an underlying framework for boundaries defined by displaced subdivision surfaces. They provide local and global shape information, combining the strengths of multiscale skeletal modeling with the multi-resolution, deformation, and shading properties afforded by subdivision surfaces. Hierarchically linked medial figures allow figural, object-based deformation, and their stability of object representation with respect to boundary perturbation shows the advantages of tolerance-based medial representations over Blum axis and Voronoi-based skeletal models.

M-reps are based on the *medial atom*, a discrete point on an implied medial locus which designates paired points (*medial involutes*) on the object boundary. Implicit in this description is a radius-proportional tolerance on the boundary positions implied by the medial atoms. These atoms are then grouped into a linear chain or planar mesh to define a single-figure m-rep, representing a discrete medial sampling of an implied solid object. Multi-figure objects may then be created hierarchically by joining figures to one another as protrusions (additive subfigures), indentations (subtractive subfigures), or associated, neighboring figures.

During my research on m-reps, I have achieved the following results:

- Shown how m-reps provide a tolerance-based skeletal technique that avoids the instabilities of Blum-style/Voronoi-based skeletal methods; this enables their

---

<sup>2</sup>**Medical Image Display and Analysis Group**—an interdepartmental research group including members of Computer Science, Radiology, Radiation Oncology, Psychiatry, Biomedical Engineering, Biostatistics, Statistics, Mathematics, Surgery, Family Medicine, and Ophthalmology.

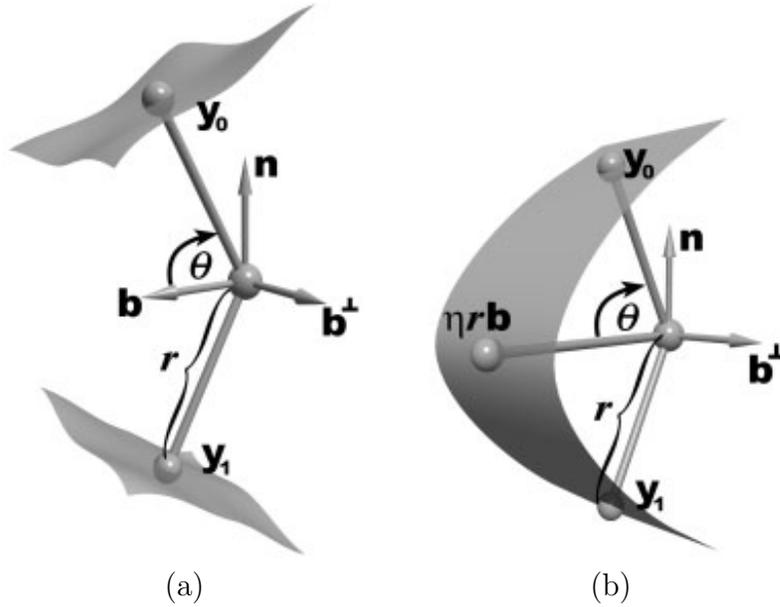


Figure 1.2: (Tom Fletcher) Example medial atom and medial end-atom. Medial atom (a) shows the frame vectors and vectors to medial involutes at the boundary. The end-atom (b) has an additional vector describing the crest region at distance  $\eta r$  along the  $\vec{b}$  vector.

use as robust shape-descriptors allowing natural shape variations to be included in models as statistical priors (in a Bayesian sense);

- Built on earlier work on sampled medial primitives in 2D and 3D to create the *medial atom*, a discrete point on an implied medial locus defined by  $m = \{\mathbf{p}, F, r, \theta\}$ , where  $\mathbf{p}$  is a point in  $\mathbb{R}^3$ ,  $F \in \mathbb{Q}$  is a coordinate frame (stored as a quaternion rotation from canonical basis, but visualized as the right-handed frame  $\{\vec{b}, \vec{b}^\perp, \vec{n}\}$ ),  $r$  is the medial radius, and  $\theta$  is the object angle in the  $\{\vec{b}, \vec{n}\}$  plane (see Fig. 1.2(a));
- Created modified medial end-atoms for the edges of the medial mesh, allowing a parameterized elongation term (see Fig. 1.2(b));
- Created data structures for representing nets of medial atoms with different topologies, including simple quadmeshes, trimeshes, linear chains, circularly connected chains, and toroidally connected quadmeshes;
- Developed techniques for deforming m-reps by operations on selected groups of medial atoms while preserving object topology;

- Explored techniques for interpolating a continuous medial axis based on the sampled medial atoms;
- Developed (with Pizer and Fletcher) a medially defined  $(u, v, t)$  coordinate system for positions on the boundary of quadmesh figures, extensible to  $(u, v, t, \tau)$  for specifying spatial locations in the coordinate system of the medial figure;
- Created a new algorithm for interpolation using Catmull-Clark approximating subdivision, thus allowing a close fit of a subdivision surface to the medially implied boundary of an m-reps; this fast interpolation technique generalizes to other stationary subdivision surfaces;
- Created proximity/nearpoint tests for the above surfaces that generalize to proximity detection on hierarchically refined stationary subdivision surfaces of other types as well;
- Explored techniques for finding true medial loci for the above-described subdivision solids;
- Implemented techniques (with Fletcher and Gash) for blending figures and sub-figures using implicit-surface methods on subdivision boundaries;
- Developed a technique (with Qiong Han) for smoothly joining figures and sub-figures, using remeshing of bounding subdivision surfaces to create a join region with user-specified curvature properties;
- Using the above technique, shown how CSG-style addition and subtraction operations can be done using figure-subfigure hierarchies;
- Demonstrated m-rep's inherent *representational polymorphism*, multiple ways to represent the same object depending on the desired operations and deformations upon it;
- Used displacement meshing of subdivision surfaces, similarly to techniques developed by Lee et al. [96], to produce boundary perturbations within the medially defined, width-proportional tolerances;
- Used image textures on m-reps and used displacement textures to produce boundary perturbations within the medially defined, width-proportional tolerances;

- Created animations (with Tom Fletcher) in realtime illustrating the natural appearance of medially based deformation of m-rep models;
- Developed *Rakshasa*, a research tool and code-base for experimenting with medial deformation and image and displacement texturing of m-reps;
- Collaborated in the development of *Pablo*, a research tool and code-base for applying m-reps to medical image segmentation of 3D computed tomography and MRI data.

I will give a brief explication of the advantages afforded by m-reps and then give a chapter outline of this dissertation.

## 1.3 Advantages of m-reps

This section provides a short overview of the advantages of m-reps that will be described in subsequent chapters; it also discusses the concept of *tolerance* as it is used in this work. There are five main advantages of m-reps over other CG modeling and rendering primitives: the multi-local nature of medial representations over local b-reps; the stability of medial representations with tolerance versus traditional Blum or Voronoi skeleta; their natural separation of object shape into intuitive, figural hierarchies; the power gained by using displacement subdivision techniques for fine-scale modeling; and the stochastic model generation possible due to m-reps' design as robust statistical shape descriptors. Table 1.1 gives a comparison of m-reps with common modeling primitives, and the following sections briefly explain each of the advantages.

### 1.3.1 Of medial primitives

The global shape information provided by the medial skeleta gives m-reps a figural basis of shape description, much more directly than in b-rep modeling, where often a skeletal structure will be *imposed* on the b-rep for purposes of articulated modeling and deformation. Skeletal methods are used commonly in modeling tools and animation work today, in such systems as Maya<sup>TM</sup> and 3D-Studio Max<sup>TM</sup>; m-reps fit into this paradigm and have advantages due to the integration of medial structure into the basic modeling primitives. M-reps thus allow medial-based figural deformation, providing natural movement for articulated models.

<i>Primitives</i>	<i>Global geometry</i>	<i>Tolerance</i>	<i>Multiscale</i>	<i>Shape statistics</i>
B-rep models	No	No	No	Yes
subdivision models	Somewhat	No	Yes	No
sph. harm. models	Only!	No	Somewhat	Yes
CSG models	Yes	No	No	No
Implicit models	No	No	Somewhat	No
convolution surface	Yes	No	Somewhat	No
Volume models	No	No	No	No
Medial models	Yes	No	Some	Yes
IBR models	No	No	No	No
M-rep models	Yes	Yes	Yes	Yes

Table 1.1: Modeling primitives and their features.

### 1.3.2 Of tolerance-based methods

The term *tolerance* has been used so far in this work without definition but requires clarification to prevent confusion with other usage. *Tolerance*, as commonly used in graphics and modeling (e.g., by Jackson [85]) describes a necessary numerical or geometric precision in modeling an object. In this sense, multiresolution surfaces, systems involving interval arithmetic for bracketing shape intersections, and even simple render-to-pixel-scale surface-spline subdivision algorithms, are all tolerance-based modeling or rendering systems.

*Tolerance*, as the term is used in m-rep research, has a broader meaning than simply “necessary numerical or geometric precision.” The notion of tolerance developed for m-reps is based on a statistical description of shape, with objects having a probabilistic nature—a mean description and a distribution of possible deviations from this mean. The statistics for a given shape may be implicit—implied by samples from a real population—or explicit, determined by a shape-synthesis methodology. For a modeling primitive to act effectively as a tolerance-based shape-descriptor, it must therefore describe not a single geometric object but a class of related objects, and it must include necessary machinery for supporting a statistical description of object shape.

It is for this reason that m-reps use a medial and figural description of object shape, supporting descriptions of classes of objects by two key aspects:

- the development of statistics on coarse-scale variability and intuitive object-shape characteristics such as thickening, bending, and elongation;
- the separation of coarse-scale shape characteristics from fine-scale details.

In this way, statistical characteristics of shapes being modeled (in geometric design) or segmented (in image analysis) can be reflected in Bayesian terms by statistical priors on the m-rep modeling primitives. The tolerance of the base primitive—which for a medial primitive is a width-proportional tolerance, see Sec. 2.3.1—is merely a reflection of the need to model the tolerances in the description of a natural object. This tolerance provides a *modeling aperture*, akin to the *sampling aperture* of shape-analytic techniques, tying the medial sampling to the shape-characteristics of the object being modeled.

M-reps allow a population-statistics approach to creating or analyzing m-rep shape descriptors, using the techniques of Fletcher et al. [55, 56] as discussed in Sec. 6.3.4. Using Lie algebra techniques, a shape can have a mean m-rep description within a population of m-reps that describe it within statistically defined tolerances. Similarly, the problem of medial atom interpolation (discussed in Ch 4) requires an understanding that there is not a single correct interpolating atom but, rather, a distribution of possible medial atoms within acceptable tolerances.

M-reps decouple coarse, large-scale shape from fine surface-detail. This provides the advantages of skeletal primitives while avoiding the instabilities of medial methods based on Blum or Voronoi skeleta, which are extremely *intolerant* of boundary noise and perturbation. Two similar shapes—in the same conceptual shape-class—may have very different Blum axes due to the instability of the boundary-to-medial transformation. M-reps present a multi-tiered approach, allowing medial description based on different medial sampling densities, using the hierarchical LODs of subdivision surfaces to provide detail within implied boundary tolerances, and finishing with a displacement texture applied to the boundary surface.

### 1.3.3 Of hierarchical object definition

M-rep models, because of their hierarchical nature, have advantages for animation and deformable modeling and such. Hierarchical modeling is nothing new in CG: hierarchical transformations are stock-in-trade in most modeling and rendering APIs, and tree-based scene-graph hierarchies are the rule in computer game design. In offering pseudo-Boolean construction operators, m-reps share some of the characteristics of CSG models in CAD application. They differ from CSG techniques in the inherent object-based deformability of the figural primitives, through bending, thickening and elongation operations applied medially and through fine-perturbations applied at the boundaries. This allows CSG approaches to be applied to the deformable shape

primitives, and the figural coordinates discussed in Ch. 3 allow the figural hierarchy to impose its own coordinates for figurally-based shape description and deformation.

### 1.3.4 Of displaced subdivision surfaces with sampled medial skeleta

Displacement subdivision surfaces are a recent development in CG modeling, coming from initial work by Lee et al. [96] as well as work by Guskov et al. [76], whose *normal meshes* are similar, though not identical to displaced subdivision surfaces. My own development work on displacement images and displacement meshes for subdivision surfaces paralleled development of these techniques: [148] from 2000 has mention of at-the-time ongoing work on displacement textures for m-reps bounded by subdivision surfaces.

Displaced subdivision surfaces are an ideal representation for m-rep boundaries; by restricting boundary displacements to be within the desired medially implied tolerances, they provide precise, fine-scale modeling of the surfaces within the framework of the coarse shape description afforded by the sampled medial skeleta. This allows coarse shape modifications to be made skeletally and independent of fine-scale detail. Tolerances must be not only width-proportional but also inversely proportional to boundary curvature to prevent local self-intersection—see Sec. 5.1.3.

Displaced subdivision surfaces—and multiresolution surfaces, in general—are exciting tools for shape modeling and design, and new developments will continue to add power to m-rep techniques and methodology.

### 1.3.5 Of primitives designed as robust statistical shape descriptors

Because m-reps were also developed as a tool for medical image analysis, they were designed to allow statistical description of shape variation across a population. Thus, an m-rep with its statistical priors (in a Bayesian sense) can be used to generate multiple instances of an object with *natural* variation in their structure. Recent work in this area is by Fletcher et al. [55, 56] and will be discussed in Sec. 6.3.4. Statistical priors may be generated by analyzing natural objects or by simply faking them. Using statistical methods, shape variations can be added to m-reps using ideas of procedural texturing, specifying stochastic *shape textures* either explicitly or automatically based on Bayesian priors.

## 1.4 Dissertation outline

This chapter listed advantages and drawbacks of current geometric modeling primitives and presented my thesis, which addresses the drawbacks through the development and use of m-reps to provide non-local boundary information based on a tolerance-based medial axis representation.

Chapter 2 provides a discussion of related work, including research on  $2D$  and  $3D$  medial axes and on skeleton-based modeling in computer graphics and CAD and also discussion of the state of the art in subdivision surface research. It also outlines other approaches to m-reps being explored in parallel with this dissertation research by myself and other members of MIDAG and associated researchers.

Chapter 3 discusses object description using sampled medial sheets, including an introduction to the theoretical and the practical ideas behind tolerance-based geometric primitives. It discusses medial coordinates and medially implied correspondence, and Sec. 3.4 discusses the ways m-reps allow topology-preserving deformations in a very natural manner.

Chapter 4 discusses creation of medially implied boundaries for m-reps using subdivision surfaces fit to a boundary polyhedron implied by the boundary involutes of the sampled medial points in the m-rep. (An alternate method, using implicit surface techniques akin to blobby modeling or convolution surfaces, is discussed in the Related Work in Sec. 2.4.3.) Techniques are developed for mapping medial coordinates onto such boundaries. Section 4.2 develops a proximity test for finding surface near-points on an interpolating Catmull-Clark boundary of an m-rep, based on the Phong-normals of the boundary polygons at different levels of subdivision. Section 4.3 develops a method for computing medial atoms and for interpolating a continuous medial axis based on locations on the subdivision boundary as specified in medial coordinates.

Chapter 5 discusses how interpolating subdivision surfaces on a medial skeleton form the basis for 3D object modeling by m-reps. It discusses the following subjects: (a) tolerance-based surface deformation using either displacement textures or displacement meshing on subdivision surfaces; (b) use of medial coordinates for locating and joining subfigures at figural boundaries; (c) remeshing for smooth joining between figure and subfigure subdivision boundaries.

Chapter 6 discusses the systems that have been created for modeling with discrete m-reps. This includes a discussion of the implementation in Rakshasa (in Sec. 6.1), a

prototype CAD tool for creating models for use in computer graphics; and the Seurat library and Pablo application (in Sec. 6.2), tools for designing m-reps of anatomical organs from 3D medical datasets and using them as deformable models for segmentation and statistical analysis. It also discusses work by others using discrete m-reps for shape analysis and solid-body deformation applications.

Chapter 7 explores the necessary conceptual shifts required to implement multi-scale modeling using m-rep hierarchies. It discusses directions for future work and promising applications for m-reps in computer graphics, CAGD, and image analysis.

# Chapter 2

## Related Work

M-rep research has drawn from a substantial body of prior work on computer graphics primitives, especially medial methods in both in CG and IA, multiscale image analysis, and multi-level subdivision surface modeling. Ch. 1 discussed some perceived drawbacks with traditional modeling and rendering primitives. In coming to a displacement subdivision, multiscale medial approach to geometric modeling, I drew upon a wealth of prior research, as well as on discussions with Stephen Pizer, Gary Bishop, Turner Whitted and others. The main ideas that form the foundation of m-rep modeling come from the following research directions:

- skeletal methods in IA: image-analysis techniques based on the Blum symmetric axis, and, conversely, model-based image segmentation based on medial methods; and skeletal methods in CG, especially those based on Blum and Voronoi skeleta;
- ideas of resolution-independence in modeling primitives based on implicit functions and volume graphics concepts;
- tolerance-based image-analysis techniques based on Gaussian scale-space notions, particularly the multiscale medial axis (*MMA* or *Core*);
- developments in subdivision surface modeling.

This chapter will discuss prior research in these areas that either influenced my own ideas or provided good indications that others saw the same drawbacks and sought similar solutions. I will, as well, discuss other directions in m-rep research at UNC which grew out of the same fertile soil as my own work but bore different fruits, with their own strengths and weaknesses.

## 2.1 Medial skeleta: the Blum symmetric axis and Voronoi-based techniques

skeleton (*'ske-l&t&n*), *noun*, New Latin, from Greek, neuter of *skeletos* dried up; akin to Greek *skellein* to dry up.—Webster's.

H. Blum, in his seminal 1967 work [16], first described the medial-axis transform as a tool for shape analysis; in later work in the 70's, he renamed it the symmetric axis transform as it included other symmetry axes beside medial ones [17, 18]. Each point on the axis gave both position and radius for a maximal circle bitangent to the object boundary; the connected loci of these points created the continuous medial axis.

In 3D, work by Nackman [106] and Nackman with Pizer [108] explored medial-to-boundary relationships for objects defined by a 2D medial sheet in  $\mathbb{R}^3$ , extending the work of Blum and Nagel and deriving curvature relationships between medial positions and corresponding positions at their boundary involutes. Most recently, James Damon [39, 40, 41] provided theoretical results on generalized offset surfaces (including Blum-axis based surfaces as a subset) in  $n$ -dimensions, deriving metric and curvature tensors and describing singular events such as local self-intersections.

Concurrent with Blum's work were the development of Voronoi region techniques (and their duals, Delaunay triangulations) in the computational geometry community; one can see that the internal Voronoi regions of a closed polygon are separated exactly by its Blum medial axis.

### 2.1.1 Geometric analysis—b-rep to m-rep versus m-rep to b-rep

There exists a long history of boundary-to-medial research in the image analysis community; hundreds of papers have been devoted to the medial-axis transform and its implementation. The amount of interest in the topic reflects the power of medial representations for understanding object geometry; the multiplicity of methods, on the other hand, reflects the fact that the boundary-to-medial transformation is *inherently unstable*. For explicit geometric models, most techniques involve analytic computation of Voronoi-based medial structure (e.g., see Culver [36]). Even slight perturbation of boundary positions can cause profound changes in such medial structures. This is especially problematic in image analysis, where any measurement noise,

pixelation error or other uncertainty in the measured boundary positions can produce wildly different medial branching structures. Fundamentally, the notion of “boundary” itself is an abstraction from and not an intrinsic property of image data. Thus, for any but the most primitive and contrived binary images, deriving a stable, semantically valid medial transformation requires an explicit regularization of the image data. This requires either (a) preprocessing to eliminate noise and boundary uncertainty, or (b) enforcing geometric relationships during processing, e.g., by a pruning process or by establishing a probability measure over the medial branches giving the degree to which they represent semantically meaningful elements of the medial skeleton (see Katz [89]).

A distinction can be drawn between image-analytic shape definition and model-based image segmentation based on synthetic models. Stephen Pizer contrasts medial image analysis (whether by Blum or Voronoi skeleta, cores, core atoms, etc.) with *reality analysis* via medial models such as statistical m-reps. The dichotomy may be likened to Plato’s Allegory of the Cave [124], wherein an unseen “ideal” object casts a “shadow” onto the “wall” of our perceptions. A 3D data-set is the shadow of a real object: image-analytic techniques treat the shadow as the object of study itself, while model-based techniques work from *a priori* knowledge based on shape properties and statistics, trying to fit known attributes of the object to the shadow being cast. In shape design, this is akin to fitting models to match desired shapes such as in parametric modeling in CAD [101]. This dichotomy may be expressed as

image-analytic	<i>a posteriori</i>	<i>deductive, data-derived</i>
model-based	<i>a priori</i>	<i>inductive, knowledge-derived.</i>

In terms of medial shape description, this expresses itself as

b-rep-to-m-rep	<i>analytic</i>	<i>unstable</i>
m-rep-to-b-rep	<i>synthetic</i>	<i>stable.</i>

The idea thus developed in the work of Pizer and his colleagues in image analysis was to reverse the medial transform, define objects medially and allow the medial structure to imply the boundaries [120]. Further, we chose a discretely sampled medial representation rather than a continuous one. This, in an analytic approach, would be seen as a geometrically lossy boundary-to-medial transformation; instead, as a synthetic, medial-to-boundary transformation, it determines the object boundary to within a sampling-proportional tolerance. This shape-synthetic approach provides the foundation on which m-reps have been constructed. Section 2.3 will discuss

this in more detail, and, in particular, why a multiscale approach is advantageous—why the boundary tolerance makes sense independent of the medial sampling and is proportional to the medial width.

### 2.1.2 Medial-primitive representations in CG

Skeletal methods in 2D or 3D have been used in computer graphics, CAD and animation for more than 25 years. In 1976, Burtnyk and Wein [22] used skeletal models for interpolation in key-framed 2D animation (*in-betweening*). Rather than being based on a Blum axis, these skeleta were instead point-mappings into a deformation coordinate system, specifying guide-points in the image for producing “rubber-sheet”-style deformations of the images. The use of medial skeleta in 3D modeling and animation has been popularized by the need, similarly, to tie motion and physical dynamics to traditional b-rep models. Surface polygons and vertices are associated with a skeletal structure that can then be used to manipulate the model using key-framed motions, physically based dynamics, or motion-capture systems. This has found widespread use in commercial modeling and animation software such as SoftImage<sup>TM</sup>, Maya<sup>TM</sup> and 3D-Studio Max<sup>TM</sup>.

J. Brandt [19] in 1992 described the use of pruned Voronoi skeleta as a 3D geometric modeling tool, based on ideas inspired by Nackman and Pizer, and by the theoretical work of Rosenfeld [128] on axial representations of shape. Brandt developed algorithms for both skeletonization and reconstruction; he acknowledged the problems of spurious axes produced by boundary noise and introduced techniques for pruning 3D axes as well as for classifying medial points based on maximal-sphere tangency conditions.

Gascuel, Verroust and Puech in 1991 proposed a method for applying a boundary “skin” to articulated skeletal primitives using a spring-network for boundary-to-medial attachment [67]. In 1993, Gascuel proposed another skeletal modeling representation based on isosurfaces of potential fields, as a tool for precise-contact computation between deformable solids [66].

Lazarus et al. [95] in 1994 developed techniques for axial deformation, extending earlier work on free-form deformation by Barr, Sederberg, et al. By first establishing a correspondence between a surface and a control axis, simple axial deformations could control boundary modification. Shapira and Rappaport [131] discuss 2D shape-blending (morphing) deformations using a star-skeleton derived from a polygonal model.

Teichmann and Teller [147] derive an articulation skeleton from a 3D polygonal model using Voronoi skeletonization and pruning, and they bind the boundary to the skeleton by a spring network. This process has been highly automated; with little user-interaction, it can import polygonalized models into a skeleton-based deformation system for dynamic modeling and key-frame animation.

The work of Storti et al. [142] on skeleton-based modeling comes the closest of any to m-reps, including using a discrete sampling of a medial-axis to determine the boundary, which can be iteratively modified according to desired object constraints on volume, inertial moments, etc. They also discuss LOD control, hexahedral mesh generation based on the medial surface for finite element modeling (*FEM*) and shape interpolation/morphing, and they use Catmull-Clark surfaces to bound their 3D models—though not interpolating ones; rather, they solve for a final desired shape by manipulating their initializing, medial-derived boundary mesh. Storti’s sampled skeletons have several important differences from m-reps:

- they use a continuous branching skeleton as an intermediate representation, not as a primary one;
- they lack a width-based tolerance on boundary position;
- they lack explicit medial frames and the shape statistics derivable from such frames;
- they lack a figural basis of shape based on linked figures and medial coordinates.

Despite these difference, Storti’s method represents a parallel approach to addressing many of the same issues as m-reps.

Blanding et al.[10] created a skeletal-based solid editor, which could import polygonal models, derive medial axes (with radius information), apply medial-based deformations and then reconstruct the modified boundary using an implicit-function approach.

Igarashi’s *Teddy* modeling system [84] fleshed hand-drawn closed figures into 3D solids by implying rotational symmetry about derived chordal axes for the figures. While it used only 2D chordal axes based on hand-sketched outlines, it showed the power of a figural-based modeling system for intuitive shape-modeling and also fleshed the 3D surfaces about the axes using (Loop) subdivision surfaces. Thus, while not directly related to m-reps, this work shared both technical and theoretical aspects.

A medial deformation technique from the volume-graphics domain was that of Gagvani et al. in 1998 on volume animation using the skeleton tree [65]. Using a volumetric skeleton tree (i.e., without polygonalization) based on a reversible thinning procedure on the distance function, a voxelized skeletal tree could be generated, transformed, and then regrown into the full volume object. This allowed physically based deformations upon objects in volume data without the expense or drawbacks that volumetric deformations would entail.

Cameron and Robb [23] use similar axial-skeleton based surface deformation methods for modeling anatomical organs in 3D data. Their driving application was interactive simulation in virtual-reality style applications, as an alternative to the slower finite-element or mass-spring models. Their 3D medial axes were generated by thinning of segmented 3D volume data, while maintaining correspondences between medial points and their boundary involutes.

Recent work by Allen, Curless and Popović [1] has derived kinematic skeletons based on range-scan data with landmarks to create articulated subdivision models of human figures allowing interpolated poses and deformation. They used normal displacements on their subdivision surfaces and used 1D linear skeletons with quaternion coordinate systems set at skeletal joints at positions where a physical joint would be located in a human figure.

Also from Curless's group at the University of Washington is the work of Capell et al. [24] on dynamic model deformation using interactive skeletons. Their techniques used piece-wise linear skeletons to accelerate the physically based (finite-element) deformation of a 3D character imbedded in a volumetric control lattice. By their use of skeletal methods, they were able to simulate the deformation of elastic solids without requiring a regular grid and avoiding the computational cost of FEM techniques.

Yoshizawa et al. [162] employ extracted Voronoi skeletons to drive deformation of surface meshes. They use mesh-smoothing based on tangent-flow on the medial axis, and use mesh-evolution techniques to eliminate local and global self-intersections of the deformed boundary meshes.

With the availability of computationally efficient b-rep to m-rep routines (e.g., the hardware-based Voronoi acceleration algorithm of Hoff et al.[81] and the advanced analytic techniques of Siddiqi et al.[136]) and with the steady research on m-rep to b-rep methods, the use of medial representations and skeletal methods in CG and IA will only continue to grow.

## 2.2 Implicit surfaces, volume modeling and convolution surfaces

M-reps base their shape description on a deformable, sampled medial skeleton, with an implied boundary that is subject to fine-scale perturbation at a resolution appropriate to the modeling or rendering task. A precursor to these ideas is the use of implicit functions and functional composition to represent deformable solids as regions bounded by isosurfaces—level-sets of the combined functions in  $\mathbb{R}^3$ .

Initial work on blobby volume models—models defined by isosurfaces of scalar fields in space—was that of researchers such as Blinn, McPheeters and the Wyvills (as already noted in Sec. 1.1) and the metaball work from Nishimura and others at Osaka University [109]. Using implicit surface representations, *soft* objects (the term used by Wyvill, McPheeter and Wyvill) could be created by functional composition of structural sub-objects and then rendered either by raytracing the isosurface or by polygonalizing the boundary. Such representations give both a coarse figural hierarchy and a resolution-independent shape-representation—the boundary can be sampled at the desired scale either (a) by the raytraced sampling frequency, or (b) by the grid-size specified for the particular marching algorithm used for polygonalization.

Scalaroff and Pentland [129] developed generalized implicit functions, wherein they could combine implicit surfaces with boundary offset fields for fine surface detailing, and other researches have worked with problems such as the parameterization of implicitly defined boundaries for texture-painting and other applications.

Convolution surfaces were developed by Bloomenthal and Shoemake [14] and continued in work by Sherstyuk [132, 133, 134]. Like other implicit models, objects are defined by level-curves of volumetric fields; the structural emphasis of convolution surfaces, however, is on the articulated skeletal elements over which the implicitizing function (a Gaussian or piece-wise polynomial approximation) is convolved. This makes them akin to m-reps, with model-deformations directly conceptualized as changes in the figures comprising the skeletal axis; similarly, Gascuel, Blanding, and Storti, as discussed above, all used a implicit surface reconstruction based on a derived medial axis. More recent work by Bloomenthal [13] has used convolution over medial skeletons to produce boundary vertex deformation for articulated models.

The *Skin* technique of Markosian et al. [102] used polyhedral skeletal primitives to generate implicit fields, which were combined into articulated figures and then polygonalized by converting the isosurface to a Loop subdivision boundary. While their

choice of polyhedral skeletal elements was somewhat *ad hoc*, their modeling technique showed many of the same advantages common to convolution surfaces and m-reps—skeletal modeling techniques based on figural elements with implicit boundaries.

Perhaps the most advanced implicit-surface modeling technique is the adaptively sampled distance field (*ADF*) of Frisken, Perry et al. [60], which they incorporated into their *Kizamu* character sculpting system [114]. Using signed distance fields that are sampled adaptively using spatial octrees, the authors created an implicit, volumetric primitive capable of precise carving, volume-data representation, multi-level complexity and LOD modeling, collision detection and so on. Fine detailing is created by precise distance offset functions applied to local regions; rendering is performed using a raytracing approach for which the octree structure gives excellent ray-traversal efficiency. ADFs easily lend themselves to “virtual clay”-style modeling systems—sculpting with distance fields has a lot in common with sculpting with continuous m-reps, without the ideas of the width-proportional tolerances. Thus, they are similar to convolution surfaces but have the LOD properties brought out by the adaptive sampling.

## 2.3 Multiscale and tolerance-based medial methods

One of the most important factors differentiating m-reps from other geometric primitives is their focus from the inception on tolerance-based, multiscale techniques for shape modeling. This should be distinguished from the multi-resolution aspect of various modeling techniques such as spherical harmonics modeling, wavelet approaches, and subdivision modeling. While these approaches allow coarse-to-fine, LOD hierarchies to be constructed, they lack a theoretical basis founded on the idea of aperture:

A necessary and sufficient aperture size for describing a shape can be based on a medially based, width-proportional description of that shape.

This concept arose from ideas in multiscale image analysis based on Gaussian scale-space theory, ideas dating back to 1962 in the work of Taizo Iijima but becoming an established research topic in the image analysis and vision community only in the 1980s. (Weickert et al. [155] give a discussion of the early history of scale-space axiomatics, including work on 2D scale spaces by Otsu in 1981.) In 1983, Witkin proposed the use of scale-space filtering, applying Gaussian convolution to a 1D image

to form a  $2D$  scale-space [159]; that is, given a function  $f(x)$ , form a  $2D$  function

$$F(x, \sigma) = f(x) * g(x, \sigma) = \int_{-\infty}^{\infty} f(u)g(x - u, \sigma)du \quad (2.1)$$

$$= \int_{-\infty}^{\infty} f(u) \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-u)^2}{2\sigma^2}} du \quad (2.2)$$

where “ $*$ ” denotes convolution with respect to  $x$ , and  $g(x, \sigma)$  is a 1D Gaussian kernel. The Gaussian was seen as ideal due to its well-behavedness vis-a-vis a number of properties, including its ease of differentiability and integrability, its *causality*<sup>1</sup>—i.e., the fact that Gaussian blurring will not create new structure in an image—and its spatial invariance. Witkin described extending the idea to  $3D$  scale-spaces for  $2D$  images, an idea which indeed led to their use in image analysis, followed by extensions to  $4D$  scale-spaces for  $3D$  images. Scale-space theory and practice was furthered by the work of Koenderink[94] and Hummel [83], Florack, ter Haar Romeny, Lindeberg, and others. A different path was taken by Perona and Malik [113] and by Grossberg [74], diverging from the equating of scale-space formation with solving the diffusion equation,  $I_t = c\nabla^2 I$ , an idea presented by both Koenderink and Hummel. Instead, they proposed to solve a non-linear/anisotropic diffusion equation

$$I_t = \nabla \cdot (c(x, y, t)\nabla I)$$

based on a variable conductance function  $c$  dependent on image structure across the image and over time. In this way, they sought to eliminate noise while preserving semantically meaningful detail such as boundaries even at coarse scales. This work was continued by many others, including Whitaker in his work on geometry-limited diffusion [156, 157].

### 2.3.1 Object description by scale-based representations

Scale-space methods in image analysis deal with problems arising from the ill-defined behavior of traditional edge-based image analysis when faced with anything besides precisely defined binary objects—i.e., any image based on grey-level data and with noise and varying intensity gradients across visually apparent boundaries. Simplistic ideas of edge-detection are replaced by *boundariness*-detectors, boundary functions  $B(x, y, \sigma_b)$  in  $\mathbb{R}^2$  or  $B(x, y, z, \sigma_b)$  in  $\mathbb{R}^3$ , which depend both on position and *scale* or

---

<sup>1</sup>a term used for this first by Koenderink.

*measurement aperture*. From this comes the idea of an image as a sampling in  $\mathbb{R}^n$  defining an  $\mathbb{R}^{n+1}$ -dimensional scale space produced by a convolution of the dataset with a Gaussian-derived measurement kernel across a continuous scale-space with scale dimension  $\sigma$ .

Along with these ideas came also a shift from boundary representations to medial ones. Influenced to a large degree by arguments based on studies of the human visual system (see Burbeck and Pizer [21], and with Morse and Ariely [20]), it was found that scale-based *medialness*-measures provide fundamentally stabler descriptions of 2D object shape than do simple boundariness-measures. Thus, rather than constructing scale-spaces based on boundary-gradient information, a medialness-kernel  $M(\mathbf{x}, \sigma_m)$  is used instead to define an  $\mathbb{R}^{n+1}$  dimensional measurement space of medialness-across-scale for positions  $\mathbf{x} \in \mathbb{R}^n$ . Loci of maximal medialness on subdimensional manifolds in the  $(\mathbf{x}, \sigma_m)$  space form the *multiscale medial axis (MMA)* or *core* of the object being analyzed; these are 1D ridges in  $\mathbb{R}^{2+1}$ , for example, or 2D surfaces in  $\mathbb{R}^{3+1}$ . The value of  $\sigma_m$  at a point  $\mathbf{x}$  along the axis provides a measure of object-width-at-scale at the point, similar to the radial value at points along a Blum medial axis. Importantly, the boundariness measurements that are used to derive the medialness function use a  $\sigma_b$  which is proportional to  $\sigma_m$ . Thus,  $\sigma_m$ , the object-width-at-scale, gives a natural sampling aperture for boundary measurements and—thinking synthetically rather than analytically—indicates a natural, width-proportional tolerance for fine surface detail.

Eberly explored the differential geometry of scale space in [49], showing that Gaussian scale spaces have a non-Euclidean, hyperbolic geometry. He also developed techniques for tracking curves of maximal medialness in such spaces, allowing the computation of cores without requiring full-image convolutions at multiple scales (as was previous required by Morse [105]). Fritsch produced parallel approaches to the same problem [61, 62] and applied MMA-based methods to problems in medical image segmentation and in x-ray portal-image registration for radiation treatment planning in radiation oncology.

The work on ridge-tracking methods for MMA-extraction led to the first practical techniques for using extracted cores as multiscale, multi-figural shape descriptors, in the work of Pizer et al. [122]. The axes thus produced differ from Blum axes in being non-branching structures; objects are represented by linked figure-subfigure or figure-co-figure combinations of simple sheet or chain axes whose boundaries are implicitly defined in a similar manner to the unions of fields defining convolution surfaces. The

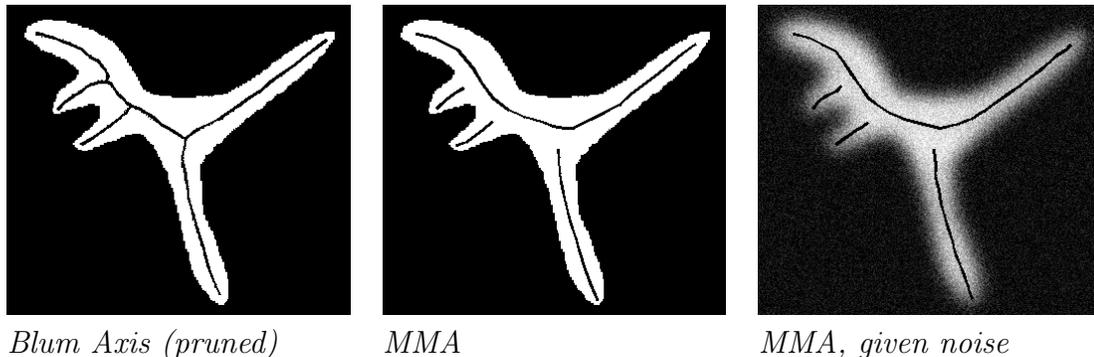


Figure 2.1: The Blum medial axis vs. the MMA. These are simulated medial axes, using the famous “Whitaker blob” to illustrate the difference between the Blum axis and multiscale medial axes extracted using methods of Fritsch or Eberly. A crucial difference between the Blum axis and the MMA is that the former requires well-defined boundaries, whereas the latter makes direct “medialness” measurements using sampling apertures proportional in size to object width. A Blum axis also requires extensive pruning of fine-scale branches to produce a medial axis describing the large-scale object shape.

width-proportional measurement apertures also make MMAs much more suited to analysis of noisy, grey-scale images, for which the Blum axis is not well-defined (see Fig. 2.1).

This work began the shift from the shape-analytic paradigm to the statistical, model-based segmentation methods for which m-reps were developed. Image structure could be segmented by medial-based primitives which inherently accounted for boundary-deviations from the norm based on the width-proportional sampling aperture about the medial axis. Segmentation thus became a process of

1. perturbing a medial model to fit an image-structure based on optimization over width-proportional “medialness” measures, and
2. perturbing the medially implied boundary according to its surface normals to match fine-scale image-structure, within width-proportional boundary tolerances.

The final shift to m-reps came with the realization that a continuous medial axis contained redundant information, due to the allowable tolerance for boundary deviation, and that it therefore could be replaced by a chain or grid (for 2D axes of 3D objects) of discretely sampled *medial atoms* (or *diatoms*) that specified location, radius, and direction of boundary involutes. Originally called *Deformable Shape Loci (DSLs)*, m-reps retain the non-branching structure of MMAs, attaching subfigures to boundary

locations implied by the parent meshes. Early work on DSLs was by Fritsch et al. [63] and by Alyson Wilson of Duke University [158].

### 2.3.2 Multiscale methods in computer graphics

Multiscale medial methods have not been applied extensively in computer graphics or CAD (though see Chen’s work in Sec. 2.4 below). Multiresolution methods, however, have risen to prominence in areas such as multiresolution surface-meshing, level-of-detail (*LOD*) modeling and model-simplification. Many of these methods are based on wavelet approaches or subdivision surfaces, which are in fact related—see Stollnitz, DeRose and Salesin [141]. Recent work on such methods includes that of Guskov on normal surfaces [76] (described below in Sec. 2.5 on displacement surfaces) and that of Carr et al. [25] on radial basis functions. Spherical harmonics provide another approach to multiresolution surface-modeling, allowing compact descriptions of surfaces with spherical topology by frequency-based decomposition, allowing higher frequencies to be added or subtracted as necessary for computational efficiency or rendering resolution. As with all Fourier-style techniques, such models lack local control for modeling operations. None of these methods are multiscale in the sense of scale-space geometry, but they endorse the usefulness of object-to-rendering-scale and LOD techniques in computer graphics.

In CAD, ideas of tolerance arise in arithmetic methods for computing accurate Boolean operations in CSG. Methods employing exact arithmetic are combined with heuristics for determining when such costly computations are necessary. One of the few solid modeling methods to explicitly include tolerance is David Jackson’s work [85] on solid-model editing with boundary tolerances.

## 2.4 Other approaches to M-reps

As explained in Sec. 2.3.1, m-reps grew out of the work on MMAs and core-based image analysis. There have been and still are other techniques based on this work that parallel the sampled medial m-rep approach or explore different directions using similar ideas.

### 2.4.1 Volume-rendering using MMAs

The chief among prior graphics applications using core-based models was the volume-rendering work of Chen [29, 28], which used core-based models for both segmentation and volume-rendering of objects from 3D medical images. A multifigure, core-based model of an organ (such as a kidney figure with an indentation subfigure) would fit itself automatically to an instance of that object in a CT dataset; a splat-based approach would then render the object at an appropriate resolution and LOD based on its screen-space projection. Using his core-based m-rep segmentation, Chen reduced the time for a splat-based volume rendering from  $\mathcal{O}(V)$ , the number of voxels in the object region, to  $\mathcal{O}(S)$ , the number of projected pixels of the visible “surface”. Chen used nets of sampled medial atoms for his basic shape representation, essentially the first implementation of sampled m-reps; he used scaled boundariness measures both to fit his medial model to the image and to define the *collar*, the region of the medially implied boundary relevant for rendering.

Chen’s work has been subsumed by current m-reps, which use displaced subdivision surfaces to both represent the medially implied surface and to describe the boundary displacements within the collar region. Chen’s approach to protrusion and indentation subfigures was effective for the volume-rendering application but insufficient for the task of multiscale modeling for CG and CAD applications. Chen’s core-based models also lacked medial-based coordinates to parameterize his medial figures.

### 2.4.2 Core atoms as fuzzy shape descriptors

Another approach to core-based primitives was the statistically based, core-atom approach of Stetten [138]. While this use was analytic and not synthetic, it utilized a statistical description of object shape based on a multiscale medial correspondence of sampled boundary points, allowing classification of object regions as sphere-like, tube-like, or slab-like to varying degrees. It is this aspect of *fuzzy* shape-description that makes core-atoms worth mentioning *vis-a-vis* related notions involving m-reps.

Stetten developed methods to determine the volume of arbitrary shapes based on boundary curvature and medial scale [139]. In a multiscale framework, this gives a fuzzy volume function, a volume-at-the-scale-of-the-core.



Figure 2.2: An m-rep figure and subfigure blended in Pablo using an implicit surface.

### 2.4.3 Implicit boundary m-reps

M-reps were conceived as a new carrier for geometric information, a new primitive for both modeling and rendering, and thought was given to directly rendering the object based on the sampled medial axis without a conversion to a polygonalized or spline-based boundary. Experiments in this direction were implemented by Fletcher [53] using an implicit function approach, treating objects as the unions of blobby medial primitives interpolated across their medial meshes. In this way, m-reps became a multiscale convolution-surface primitive, being renderable by either ray-casting or a surface-marching polygonalization.

Implicit m-reps have the advantages of other implicit surface methods, especially the ease of creating additive and subtractive subfigures, joining figures and subfigures, and the resolution independence of the model. Their disadvantages lie in control of blends, the awkwardness of going from medial to implicit to b-rep, and their lack of a medially implied parameterization for texturing, boundary offsets and correspondence. Implicit surfaces were used initially in *Pablo* for figure-subfigure blending of m-reps, based on a blending function developed by Pizer and Fletcher and implemented by Gash and Thall (see Fig. 2.2); they have since been replaced by subdivision surface remeshing code developed by Thall and Qiong and described in Sec. 5.2.

### 2.4.4 Continuous medial surface methods

M-reps are constructed of meshes of discrete medial atoms, each giving position, radius, and object angles as a sampling of the object geometry. Work has also been done on 2D and 3D shape synthesis using continuous medial representations, based on fitting splines to the medial information and constructing a continuous Blum

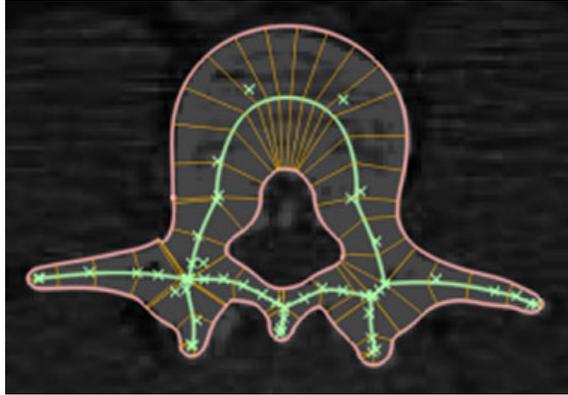


Figure 2.3: [Paul Yushkevich] A 2D continuous m-rep (*cm-rep*) of a vertebra in cross-section, based on a b-spline fit to the medial structure.

representation for figural geometry. Initial work in the *Seurat* library (see Sec. 6.2.2) and by Fletcher [54] used 2D Bezier patches for the medial surface and radius function; Fletcher also worked out mathematical “legality” checks for determining when the  $\nabla r$ -interpolation produced an illegal boundary—i.e., one with cusps, folds, or other local self-intersections. A medial interpolation function was also used by Crouch [33] in her m-rep-based FEM research.

Working with the same ideas, Yushkevich [163] developed a *continuous m-rep* (*cm-rep*) based on a b-spline interpolation of the medial surface and  $r$ -function, using the same legality checks as Fletcher, as well as ideas from Damon [38]. While 3D cm-reps are limited at present to single-figure models, 2D cm-reps are fully realized, multifigure-modeling primitives. Fig. 2.3 shows a 2D cm-rep created for a cross-section of a vertebra.

Continuous m-reps have the advantage over discrete m-reps of a true Blum-medial correspondence between the medial surface and the boundaries generated by the offset radius and directions at each point on the axis. Consequently, cm-reps suffer from the overprecision and intolerance associated with the Blum axis; legality checks are much more important to preserve boundary-legality than for discrete m-reps, where boundary smoothing by the subdivision, within tolerances, serves to prevent most such occurrences. In the view of m-rep purists (i.e., this author), Blum-based continuous m-reps simply store too much information at the medial locus, which should restrict itself to the coarse-scale shape information and store the fine detail—including smoothness—at the boundary where it belongs. The large-scale smoothness of an object considered with a width-proportional sampling aperture, as provided by

a discrete m-rep boundary, must not be confused with the fine-scale smoothness of a highly polished object. A cm-rep fails to make this distinction.

## 2.5 Subdivision surfaces

Subdivision surfaces have become one of the most powerful and useful tools in the 3D modeling toolbox. In generalizing spline-based surfaces for meshes of arbitrary topology, they allow easy fitting of (almost everywhere)  $\mathcal{C}^1$  or  $\mathcal{C}^2$  surfaces to a polygonalized mesh. Subdivision surfaces are now being added as primitives in inexpensive graphics hardware and are supported in modeling tools such as Maya<sup>TM</sup> and in graphics APIs such as DirectX 9 and OpenGL Performer.<sup>2</sup>

### 2.5.1 Interpolating and approximating surfaces

The earliest works subdivision surfaces were the face-splitting Doo-Sabin method and the vertex-splitting Catmull-Clark method from the late 1970s, both of which gave almost everywhere  $\mathcal{C}^2$  limit surfaces [26, 47]. Both of these were approximating subdivision methods—they did not interpolate their initializing control mesh—and both were generalizations of bicubic splines from regular quad-grids to arbitrary meshes. In the late 1980s, Charles Loop produced another approximating subdivision method for triangle-based meshes which generalized triangular box-splines [99].

Interpolating subdivision methods have been developed subsequently, whereby the limit surface passes through the initializing mesh-points. Dyn et al. [48] produced a *butterfly* method for interpolating triangle meshes; Hoppe et al. [82] produced a modified Loop subdivision for reconstruction purposes; and Halstead et al. [77] constructed a modified Catmull-Clark interpolating subdivision, using a bending-energy minimization to eliminate rippling effects in the interpolating surface. Of these methods, the Halstead technique approaches most closely the requirements for m-rep modeling, including normal interpolation; this will be discussed in more detail in Ch. 4. Most recently, Litke et al. [97] produced an interactive, quasi-interpolating approach using Catmull-Clark surfaces; the iteratively interpolating approach for m-reps, described in Ch. 4, is similar to that of Litke.

One other idea worth mentioning is the research of McDonnell et al. [103] on virtual clay using *solid* subdivision models. This builds from earlier work by MacCracken

---

<sup>2</sup><http://www.sgi.com/software/performer/announce.html>

and Joy [100] where they extended Catmull-Clark subdivision surfaces to subdivision-based deformation based on a 3D subdivision lattices. While MacCracken used these subdivision lattices as tools for specifying spatial deformation, McDonnell developed them as modeling primitives in their own right.

### 2.5.2 Parameterization

Many applications for m-reps require a boundary parameterization, one based ideally on an associated medial-parameterization. For many years, it was not believed possible to establish a closed-form parameterization of surfaces defined by stationary subdivision algorithms. The work of Jos Stam [137] changed this notion, describing a way to parameterize stationary subdivision surfaces (such as Catmull-Clark and Loop surfaces) based on an eigenanalysis of surface behavior in the neighborhood of extraordinary points; elsewhere, the surfaces are simple spline surfaces and can be parameterized accordingly. The work of DeRose et al. [43], more simply, presented the idea of splitting texture parameters according to the same surface-subdivision rules for vertex-averaging to create a smooth parameterization of a subdivision surface. It will be seen in Ch. 3.1.5 that a medial parameterization can be derived straightforwardly from the sampled medial axis and be extended to the solid volume it defines; Ch. 4.1.6 will discuss how these coordinates can be usefully associated with the subdivision surface bounding an m-rep, by methods similar to DeRose's.

### 2.5.3 Displacement subdivision surfaces

Displacement subdivision surfaces are very recent ideas, though their foundation in displacement mapping goes back the original texture-mapping work of Catmull in the early 70s. Using the various means of parameterization available for subdivision surfaces, it follows naturally to map boundary displacements to them to provide the fine detail that otherwise is eliminated by the inherent smoothing of the subdivision process. Thus, my own ideas (as described in Ch. 5) were developed in parallel with the work of Lee et al. at Princeton [96] and the work of Guskov on normal surfaces [76]. The Guskov paper—mentioned above in the context of LOD modeling—actually describes a more general class of hierarchical meshes; it specifies vertices with only a single scalar displacement rather than by  $(x, y, z)$  positions, with successive displacements being applied hierarchically from a base-mesh. Guskov et al. regarded the DS-surfaces of Lee as a bi-level version of their multiresolution approach; m-reps

can be thought of similarly.

Some interesting work on more general boundary displacements (not restricted to the normal direction) is that of Kobbelt et al. [93] on feature-based surface extraction from volume data. In this work, they create an enhanced distance field representation for their marching cubes algorithm, with applications to Boolean operations on volume data for CAD purposes. This is not really a subdivision approach and is more an analytic than a synthetic tool, but the work is worth mentioning for its aspects of LOD modeling, displacement fields and Boolean shape composition. Recent work by Ju et al. [88] and by Varadhan et al. [152] extend and improve Kobbelt’s algorithm.

#### 2.5.4 Boolean operations on subdivision surfaces

As an essential part of figural modeling using m-reps, this thesis promotes the suitability of subdivision surfaces for Boolean—or approximate Boolean—operations for solid modeling. This is another recent development in subdivision surface modeling, with much research from the Multi-Res Modeling Group at CalTech.<sup>3</sup> The key publication in this area is the SIGGRAPH paper of Biermann et al. [8] on approximate Boolean operations on subdivision surfaces, which drew on concurrent work by Litke et al. [98] on trimming subdivision meshes. Biermann developed techniques for additive and subtractive operations to minimize the size and optimize the quality of the new control meshes, employing a multiresolution-mesh-based approach with adaptive remeshing and using the concept of a *parametric domain of a surface* for their cutting curves; this parametric domain, set by the initial triangularization for their control mesh, is directly paralleled in the  $(u, v, t)$  coordinates that m-reps establish on their boundaries. See Sec. 5.2.3, where, as will be seen, Biermann’s use of cutting curves in these parametric domains is paralleled independently by our use of curve-dilation in  $(u, v, t)$  in the m-rep boundary remeshing scheme.

The finding of subdivision surface intersections remains an open problem—for example, see work by Grinspun et al. [73] on detecting subdivision surface interference, including self-interference. The remeshing of subdivision surfaces in join regions also presents challenges. Igarashi’s Teddy application [84] did such stitching for non-interpolating surfaces, using ideas from Barequet and Sharir [6]; older work on mesh-stitching goes back at least as far as the dynamic programming algorithm of Fuchs et al. [64] for reconstructing surfaces based on contours.

---

<sup>3</sup><http://www.multires.caltech.edu/>

The Markosian *Skins* paper [102] also deserves mention, since it blends protrusion figures based on the subdivision surface generated by distance fields about the skeletal polyhedra. It is thus an implicit-surface technique that uses subdivision surfaces as boundary interpolants.

### 2.5.5 Physically based operations on subdivision surfaces

As subdivision surfaces become ubiquitous in computer graphics and CAGD, the need to provide physically based operations on objects enclosed by subdivision surface boundaries will be expanding likewise. The most relevant current work in this area is that of Jörg Peters and Ahmad Nasri, on “Computing Volumes of Solids Enclosed by Recursive Subdivision Surfaces” [116]. This work was in turn based on that of Gonzalez-Ochoa, McCammon and Peters [72], which presented the initial theory for objects enclosed by piecewise-polynomial surfaces, from which the stationary subdivision surfaces generalize. The authors show that it is possible to do rapid, on-the-fly computation of volume and higher-order inertial moments for subdivision-surface-bounded solid; such techniques should be applicable to m-reps under medial deformation, as will be discussed in Ch. 7.3.2. The work by Stetten cited above, on determining the volume of arbitrary shapes based on boundary curvature and medial scale, is also relevant to physically based m-rep modeling, as is the FEM research of Crouch.

## 2.6 Concluding remarks on prior work

In this chapter, both direct and “spiritual” ancestors of m-reps were discussed, as drawn from a wealth of theory and practice in image-analysis and model-based image segmentation, skeletal modeling in computer-graphics modeling and CAD, and current research in subdivision-surface modeling. M-reps, as it will be seen, are a synthesis of trends found in each of these research areas and are well-poised for applications in all of them.

The next chapter will define the tolerance-based medial atom, showing how discrete meshes of these atoms form the basic primitives for shape modeling by m-reps.

# Chapter 3

## Object Description by Sampled Medial Sheets

Section 2.1 discussed the Blum symmetric axis and its relationship with Voronoi skeleta. Blum and Voronoi axes form the basis of most methods for skeletal modeling and reconstruction. A key observation of multiscale medial research, however, is that Blum medial methods are *overprecise—intolerant*—in describing object geometry and that they pay for this intolerance with their sensitivity to boundary perturbation. This is inevitable: because the Blum axis representation is isomorphic to the boundary representation, all boundary perturbations map directly to branches of the medial skeleton [18]. An object with fine surface detail must therefore map to a bushy skeletal tree; minute changes in the boundary may produce major changes in the skeletal structure, and any pruning of this tree produces a loss of related surface detail. Such pruned skeleta thus produce a lossy compression of the shape-related geometric information.

M-reps are founded on the idea that, for a shape-synthesis task, the information loss can be seen in reverse as rather a feature than a bug—i.e., rather than *settling* for a lossy geometric representation, we insist upon one. M-reps replace Blum- and Voronoi-axis-based methods by one with a multiscale medial axis as its theoretical foundation. By assuming from the start that a stable skeletal representation requires insensitivity to fine surface detail, m-reps divide the shape characteristics of an object into two parts:

- a coarse-scale, medial representation based on object width but with a width-proportional boundary tolerance, and

- a fine-scale, boundary representation giving perturbations within the width-proportional tolerances at each surface location.

The idea of sampled medial sheets as modeling primitives grew out of these realizations, driven by the need for a multiscale representation for image-analysis based on geometric models. The initial ideas for a sampled medial representation belonged to Stephen Pizer, and most of the critical ideas of sampled m-rep modeling can be variously credited to Pizer jointly with members of his research group, including Fritsch, Wilson, Chen, and me; my own contributions included the following:

- creating data-structures for discrete 3D medial atoms with explicit quaternion-based medial frames;
- creating sampled medial sheets of various topologies and defining medial-based deformations upon them;
- developing the  $(u, v, t)$  parameterization for quad-meshes with planar connectivity and extending this to a  $(u, v, t, \tau)$  parameterization of the space inside and outside the m-rep;
- exploring boundary interpolation based on medial interpolation, on polygonal boundary tessellation, and ultimately on the displacement subdivision boundaries that are used in m-reps in their present form.

These ideas have been implemented in Rakshasa—an experimental platform for medial shape creation and boundary deformation experiments—and in MIDAG’s Pablo project; these implementations will be discussed in detail in Ch. 6. This chapter will discuss the following topics: medial atoms and the sampled medial sheet; medial-based coordinates; figural hierarchies; medial interpolation; and topology preserving, medial-based deformations. It will conclude with a discussion of drawbacks and limitations of sampled medial representations, both theoretical and as currently implemented in m-reps.

### 3.1 The sampled medial sheet

The structural unit of the m-rep model is a single-figure medial mesh made up of medial atoms linked in chains or grids. The atoms, as discussed in the previous chapters, provide a width-proportional sampling of the medial structure and describe discrete

boundary involutes within a width-proportional tolerance. This section describes (a) the medial atom with its explicit, medially derived coordinate frame, (b) the single-figure medial mesh, (c) the invariance of m-reps under similarity transformation, and (d) the parameterization by medially derived coordinates which provides correspondence between the implied medial axis and the implied single-figure boundary. This parameterization can be extended to a parameterization of space in the neighborhood of the medial figure. The section concludes with discussion of the central concept of width-proportionality as introduced in Sec. 2.3.1.

### 3.1.1 The medial atom

The medial atom represents a Blum-style sampling from a smoothed boundary, a boundary viewed at a width-proportional scale—Blum-like in including  $\nabla r$  information in the medial representation. Other approaches to medial atoms (notably Stetten’s) have used a chordal axis, where surface-normals at boundary involutes are not parallel to the involute-vectors themselves. Individual medial atoms in these chordal representations contain only  $0^{th}$  order boundary information, however—they store no information on the change of the medial radius-function in the neighborhood of the medial atom.

In implementation, medial atoms contain medial position, radius, object angle, and medial frame information. Formally, a medial atom is a discrete point on an implied medial locus defined by  $\mathcal{M}_{\mathbf{p}} = \{\mathbf{p}, F, r, \theta\}$ , where  $\mathbf{p}$  is a point in  $\mathbb{R}^3$  on the implied medial manifold  $\mathcal{M}$ ,  $F \in SO(3)$  is the medial frame orientation with respect to a canonical basis,  $r$  is the radius, and  $\theta$  is the object angle (see Fig. 3.1).

The object angle  $\theta$  is found by the relationship

$$\cos \theta = \|\nabla r\|$$

where the gradient is with respect to arc-length on the medial surface along geodesics through the point  $\mathbf{p}$ .

The frame orientation  $F$  is represented by a unit-length quaternion  $q \in \mathbb{Q}$ . Unit-length quaternions are ideal for specifying frames in  $\mathbb{R}^3$  relative to a canonical basis-frame, due to their providing a double-cover of the space of rotations  $SO(3)$ . They have well-established advantages over rotation matrices for representing orientation.<sup>1</sup>

---

<sup>1</sup>Their only drawback is the need to transform them to a traditional rotation matrix to multiply them into a composite transformation matrix pipeline in homogeneous coordinates. They work well

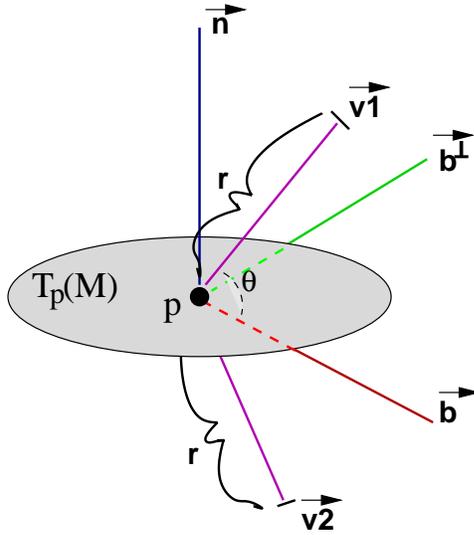


Figure 3.1: A medial atom at position  $\mathbf{p}$  and its frame  $\{\vec{b}, \vec{b}^\perp, \vec{n}\}$  relative to the tangent space of the medial surface,  $T_p(\mathcal{M})$ .

(See Shoemake [135].) The frame, as shown above, orients the vectors  $\vec{b}$  and  $\vec{b}^\perp$  in the tangent space  $T_p(\mathcal{M})$  of the implied medial surface  $\mathcal{M}$  with  $\vec{n}$  normal to it;  $\vec{b}$  is in the  $-\nabla r$  direction in  $T_p(\mathcal{M})$ —the direction of maximal narrowing of the object *vis-a-vis* its medial surface—and thus  $\vec{b}$  and  $\theta$  together give

$$\nabla r = -\vec{b} \cos \theta, \quad (3.1)$$

the established Blum relationship in  $3D$ . When  $\mathbf{p}$  is a critical point in  $r$  and thus  $\|\nabla r\| = 0$ , the boundaries are parallel at the involute positions  $\mathbf{p} + \vec{v}_1$  and  $\mathbf{p} + \vec{v}_2$ ;  $\vec{n}$  is then in the  $\vec{v}_1$  direction and is both the medial and the boundary normal. In this case, the choice of  $\vec{b}$  direction is ambiguous and can be made arbitrarily in  $T_p(\mathcal{M})$  to suit the needs of a given modeling or deformation task. The existence of a coordinate frame at each medial atom allows medially based deformations to be specified relative to a particular atom; such deformations will be discussed in Sec. 3.4.

There are special-case constructs for *end-atoms*—those at the edge of the sampled medial lattice. The edges of the medial sheet generically imply the *crest* regions of the implied boundary, regions where osculating spheres of radius  $r$  may have higher-order contact with the boundary than the bitangency of atom involutes in slab-like regions, though not necessarily at the sampled atom positions. The special end-atoms which

---

in an OpenGL-style rendering pipe, however, since they store rotations in vector-angle form ideal for input.

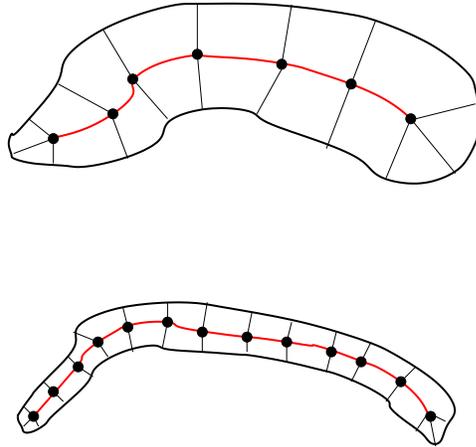


Figure 3.2: A width-proportional sampling of a fat and a thin object. No exact constant of proportionality is used; the sampling reflects the inherent medial deformability of an object region based on the width vs. the distance along the medial axis.

deal with crest regions will be discussed in the next section.

### 3.1.2 The single-figure medial mesh

Single-figure m-reps in  $3D$  are made up of  $1D$  chains or  $2D$  lattices of medial atoms. For Blum medial axes,  $1D$  medial curves are non-generic primitives—any boundary perturbation may cause the  $1D$  medial structure to split and generate  $2D$  medial regions represented flattened regions along the tube-like boundary. For m-reps, however,  $1D$  chains are generic primitives; boundary perturbations within tolerance do not affect the medial structure. The sampling density of the medial atoms in the chains or lattices should be inversely proportional to the object width, in keeping with the medial deformability required by the model. Fig. 3.2 shows how coarse deformation of the model at the medial level is constrained by the object width—a fat object has less medial variability and requires fewer samples than a thin object of equal length. (Intuitively, a snake can be wigglier than a knockwurst.)

Width-proportional sampling is akin to the radius-of-curvature-proportional sampling required for boundary reconstruction, as, for example, in the work of Amenta et al.[2] (where the authors also use an approximate medial axis to guide reconstruction). Radius-of-curvature-proportionality is a bad sampling choice for an m-rep, however, because the sampled medial skeleton needs to model the coarse-scale *variability* of an object with the given widths. Thus, a sparse medial representation for a long,

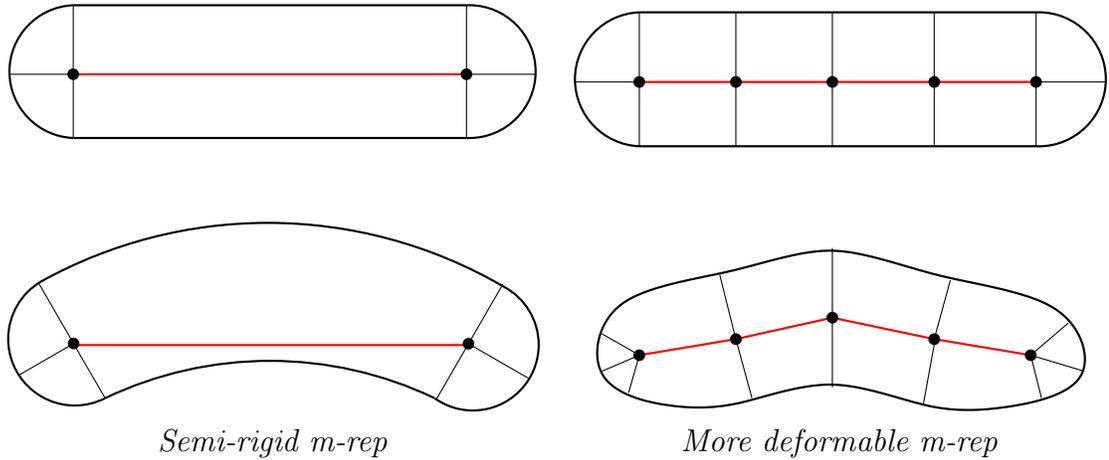


Figure 3.3: Two m-reps allowing varying medial deformation, based on sampling density.

straight object would fail to model the possible curves and wiggles of the medial structure which might be encountered, as in Fig. 3.3. If an object is known to be rigid (e.g., bricks, which seldom wiggle), a sparse sampling can be used; conceptually, this varies the constant-of-proportionality of the width-based sampling to accommodate the desired medial “stiffness”. Another alternative is to adaptively resample the medial surface based on an underlying continuous model when the need for curvature variability may change across a sample-space or across time. This approach is taken in work by Yushkevich [163] on continuous m-reps (*cm-reps*) mentioned in the previous chapter. Such a resampling is also possible with discrete m-reps by interpolating new medial atoms based on boundaries in regions needing more variability; this method will be discussed in Sec. 4.3.

The dependence of the medial sampling on the expected modes of deformation for an object provide a description of the tolerance of the object being modeled, as discussed previously in Sec. 1.3.2. The sampling for a model will thus be based on average  $r$  values, the base- $r$ s for the shape, and the relative spacing of medial samplings will be held unchanged thereafter for that class of object; this is central to the way m-rep-based segmentation and shape-analysis is carried out, with deformations being quantifiable in terms of deviations from the base model (see Sec. 6.2.1).

The most commonly used m-rep in our current research is based on a regular quadmesh of medial atoms. Interior atoms represent medial position, positions and normals at a single pair of boundary involutes, and also the object angle and medial frame, giving the local orientation of the medial surface and the behavior of  $r$  and  $\nabla r$

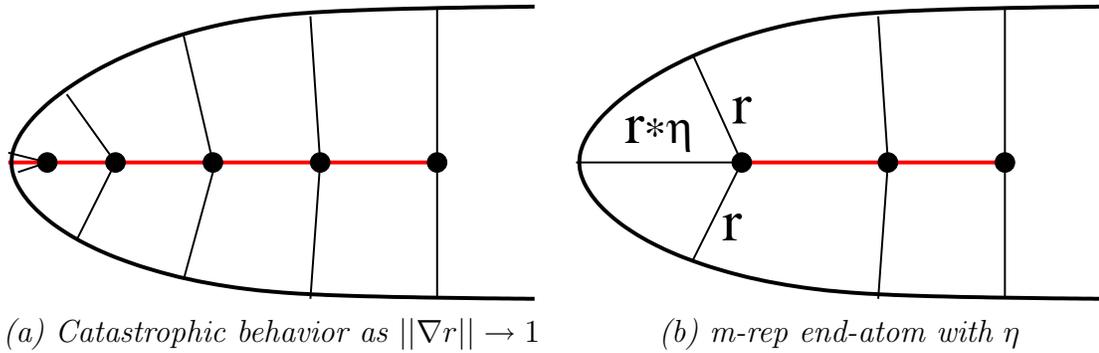


Figure 3.4: Catastrophic behavior of  $\vec{v}_1$  and  $\vec{v}_2$  in crest region at the edge of medial sheet (in cross-section).

in that region. Edge atoms are modified to reflect the behavior of the medial manifold at edge of medial sheet. In these *crest* regions, in the Blum case, as a medial point approaches the crest along the medial surface, there is a catastrophic collapse of the  $\vec{v}_1$  and  $\vec{v}_2$  vectors to the  $\vec{b}$  as  $\|\nabla r\| \rightarrow 1$ . (This is described by Giblin et al. [69] and Damon [38].) See Fig. 3.4(a). The crest region is a maximum of curvature for the object, and the cusp in the loci of boundary centers-of-curvature lies along the edge of the medial sheet. Rather than attempting to model the behavior of the  $\vec{v}_1$  and  $\vec{v}_2$  vectors in the neighborhood of this cusp, *end-atoms* are placed which assume distinct  $\vec{v}_1$  and  $\vec{v}_2$  vectors with object angle  $\theta$  from  $\vec{b}$ , and with a circularly cross-sectioned endcap connecting them. An elongation factor  $\eta$  can then be used to specify endcap deviation from this circular cross-section, as shown in Fig. 3.4(b). This also partially eliminates the problem given width-proportional sampling of infinite sampling density where the width  $r \rightarrow 0$ , as discussed below.

Pablo currently includes the elongation factor  $\eta$  for crest atoms, though the degree of deviation of the cross-section from circularity is determined automatically by the subdivision algorithm used for surface fitting to the medial skeleton. This is described in Ch. 4.

While the  $\eta$ -parameter can vary the shape of a crest to approach a sharp point, the medial skeleton of an m-rep alone implies an inherently blobby object—the boundary represented is intended as an approximation to a mean density isosurface as sampled at an  $r$ -proportional scale and tolerance. Boundary details such as corners, creases, and other first-order discontinuities have been “blurred away” by the fuzziness of the representation. This prevents the branching of the medial surface as in the Blum case. It also prevents sampling problems at corners: the medial structure of a corner has problem of infinite regress for medial sampling at  $r$ -proportional scale; as  $r \rightarrow$

0, the sampling density increases without bound. Thus, while there can be coarse representation of corners by medial atoms and elongations along crests, the fine-scale surface-detail in an m-rep is achieved at the boundary level. This can be (a) by boundary perturbation techniques such as displacement maps or (b) by adding creases, corners, and cusps to subdivision surfaces as by techniques of DeRose et al. [43]. Such methods will be discussed in detail in Ch. 5.

### 3.1.3 Invariance under similarity transformation

An m-rep of a solid object is invariant under similarity transformation: rotation, translation, and uniform scaling of an object is equivalent to the same rotation, translation, and uniform scaling of its medial representation. Similarity transformation thus preserves the relationship between the boundary involutes and their corresponding medial positions—this is a property of medial axes in general and one of the cardinal virtues of m-reps as originally proposed by Pizer et al. for image-analysis [120]. Consequently, object shape, if based on local figural relationships defined by medial shape descriptors, is similarity-transform invariant. Let  $\mathcal{T}$  be a similarity transformation  $(s, R, T)$  for  $s \in \mathbb{R}^+$ ,  $R \in SO(3)$ , and  $T \in \mathbb{R}^3$ , where  $\mathcal{T} : \mathbb{R}^3 \rightarrow \mathbb{R}^3$  as  $\mathcal{T}(\mathbf{x}) = s(R\mathbf{x}) + T$  for  $\mathbf{x} \in \mathbb{R}^3$ . Then the equivalent transformation for  $\mathcal{M}_{\mathbf{p}} = \{\mathbf{p}, F, r, \theta\}$ , an atom on the medial locus, is

$$\mathcal{T}(\mathcal{M}_{\mathbf{p}}) = \{\mathcal{T}(\mathbf{p}), R \circ F, s \cdot r, \theta\}.$$

A similarity transformation thus transforms the medial location, rotates the medial frame, and scales  $r$ , while leaving  $\theta$  unchanged.

M-rep behavior under affine transformation is problematic. Blum medial topology is *not* invariant under affine transformation; Boundary-point pairs that are mutual involutes may no longer be so after a skewing or non-uniform scaling. Thus, for a discrete medial atom, there is no simple transformation of the  $\{\mathbf{p}, F, r, \theta\}$  structure corresponding to a given non-uniform scaling of the boundary.<sup>2</sup> This means that m-reps cannot be naively manipulated by free-form deformation techniques, since the medial structure of the undeformed object does not correspond to the medial struc-

---

<sup>2</sup>In differential geometric terms, this is because the boundary positions transform contravariantly under affine transformation while the boundary normals transform covariantly. Thus, in computer graphics, when a matrix transformation  $\mathcal{M}$  is applied to surface positions, the inverse-transpose  $(\mathcal{M}^{-1})^T$  is applied to surface normals [59].

ture of the object after deformation. Although some warpings might be acceptably within shape tolerances for a particular m-rep, m-reps are in general not amenable to free-form deformation. One can attempt to modify a prior medial structure to match a deformed object (as done in Pablo), or one can fit a new medial structure to a non-uniformly transformed boundary. Conceptually, however, the “correct” deformations of an m-rep are medial deformations—width-changes and bendings based on alteration of the medial-atoms and their embedding mesh—and boundary displacements within tolerance (see Sec. 3.4.1).

### 3.1.4 Connectivity and topology of sampled medial meshes

In implementing discrete m-reps, there is latitude in the choice of mesh topologies and mesh connectivity. Section 3.2.1 discusses the inherent polymorphism in figural representation by medial representation—particular structural topologies can be generated by different figural representations, making the need for a particular structuring primitive somewhat contingent. Primitively and locally, medial structure can be categorized according to dimensional symmetry as spheres (symmetric about a point), cylinders (symmetric about a 1D axis), and slabs (symmetric about a 2D sheet) [140]. Sampled medial structures must model these symmetries as manifest in all manner of solid shapes if they are to be robust geometric primitives. My work on m-reps has focused mainly on quadrilateral meshes, but it has also explored the following mesh topologies:

- *sphere-figure*—the simplest medial figure, with a single medial primitive and with radius-proportional boundary tolerance;
- *tube-figure* and *ring-figure*—single chains of medial atoms with rotational symmetry about a 1D space-curve locus, forming solid tubes or tori;
- *quad-figure*, *tri-figure*, *slice-figure*—2D medial meshes of regular quadrilaterals or triangles modeling slab-like symmetry; slice-figures are regular quad meshes with rows of medial atoms constrained to lie within planes;
- *hollow-tube-figure*—a 2D quad mesh with cylindrical connectivity, creating a hollow tube with walls deformable proportionally to their thickness;
- *hollow-sphere-figure*—modeling a sphere as a hollow shell of radius  $2r$  by a (triangle-based) tessellation of a sphere by medial atoms of radius  $r$ .

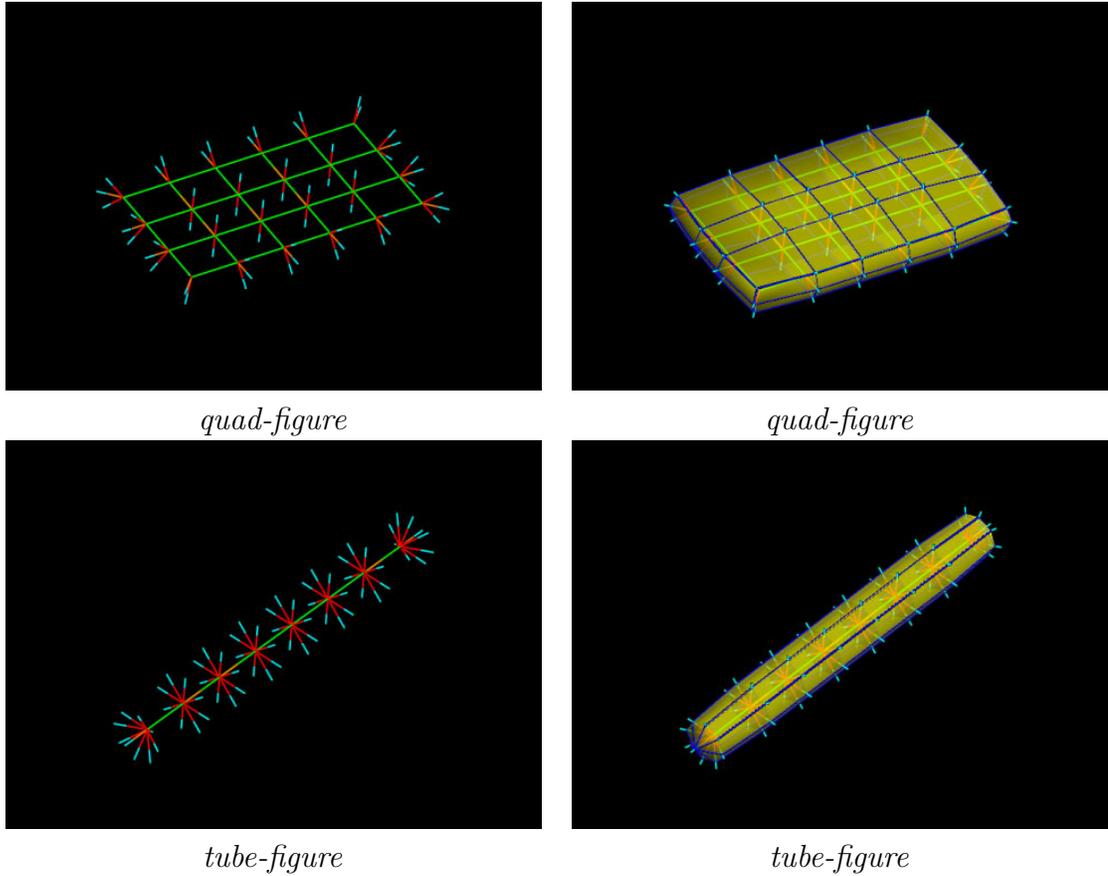


Figure 3.5: Examples of medial mesh topologies. On the left are the meshes of medial atoms; on the right are the coarse polygonalized boundaries. Only the quad-figure is in active use in current m-rep research.

Figs. 3.5 and 3.6 show examples of the most important of these. The following are some particular points about these representations.

- A sphere-figure is a generic object: unlike its analogue in Voronoi-based skeletal methods, a spherically symmetric figure with a single medial atom is stable under small boundary perturbations. Its surface may be parameterized by  $[\phi, \theta]$  or by quaternion frames giving both position and rotation for subfigure placement. As with other figures, surface displacements are in the normal directions.
- Slice-figures have the drawback of modeling only the in-slice 2D medial structure and not the 3D medial structure of the object, thus limiting the types of figures that could be modeled. They were used in early prototype applications of 3D m-reps in medical image segmentation for two reasons: (1) they could be superimposed on a corresponding plane of volume-image data, allowing one

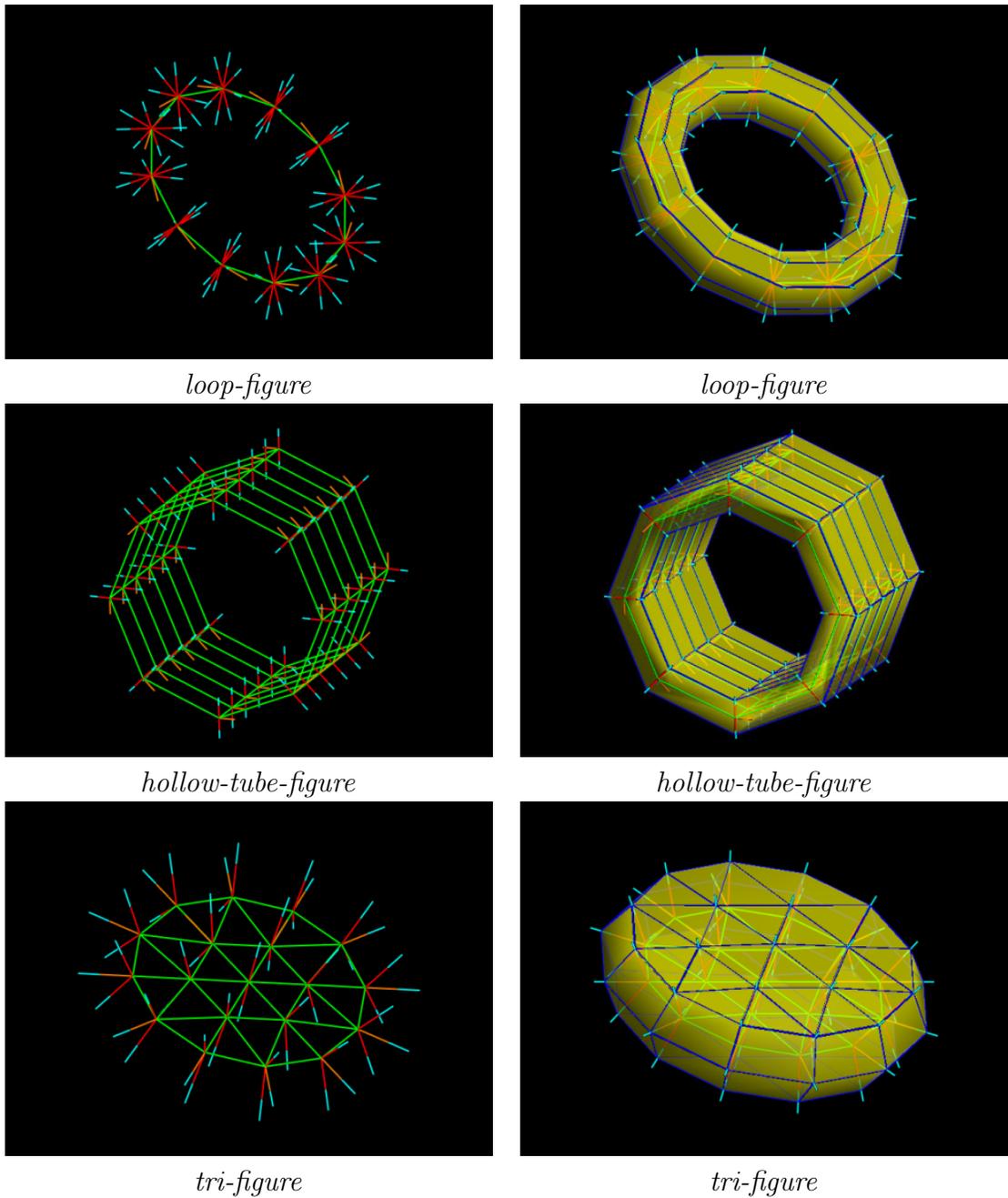


Figure 3.6: More examples of medial mesh topologies. On the left are the meshes of medial atoms; on the right are the coarse polygonalized boundaries. The tri-figures were from early experiments with non-quad-based medial mesh connectivity.

to visualize the match of M-rep to the underlying image; (2) they allowed the design of objects formed by a generalized extrusion in which a deforming medial-curve sweeps through space, possibly with twisting.

- Tri-figures provide a triangular mesh for the medial sheet; triangles were used rather than a regular quad mesh to simplify adaptive resampling of meshes, with an eye toward future work on interactive modeling and shape deformation.

Pablo currently supports quad-figures and tube-figures, which suffice for most modeling tasks. Chapter 6 discusses implementation details.

### 3.1.5 Medial coordinates and correspondence for quadmeshes

A Blum medial description provides correspondence between positions on the medial surface and their associated boundary involutes. Given a continuous, parameterized medial surface, one might then extend the medial surface parameterization to associated involute positions on the boundary. This presents difficulties in situations that we define as *non-Blum*—where there are folds or cusps in a continuous medial representation and thus no uniquely or well-defined involute relationships between boundary positions and points on the continuous medial axis, as shown in Fig. 3.7. This will be discussed below in Sec. 3.1.6. This is another case where the *intolerance* of exact representation creates problems. Because discrete m-reps have no underlying continuous representation, a parameterization is possible which respects known involute positions and approximates involute relationships between boundary positions and implied medial positions elsewhere.

A discrete m-rep quad-figure can parameterize its boundary in terms of a medially based coordinate system, and this parameterization may be extended to the interior of the object and to a local neighborhood of the exterior. M-rep medial coordinates  $(u, v, t, \tau)$  were developed by Pizer, Fletcher, and Thall for parameterization of the single-figure m-rep models being used for medical image analysis [87, 123] and were later extended to allow parameterization of the blend regions between figure and subfigure.

Given a quadmesh of medial atoms  $\mathcal{M}_{ij}$ , where  $(i, j) \in [1, N] \times [1, M]$  are the integer-valued indices for the  $N$  rows and  $M$  columns of mesh atoms, the correspondence between a medial position  $\mathbf{p}_{ij}$  and its boundary involutes  $\mathbf{v}_1 = \mathbf{p}_{ij} + \vec{v}_1$  and

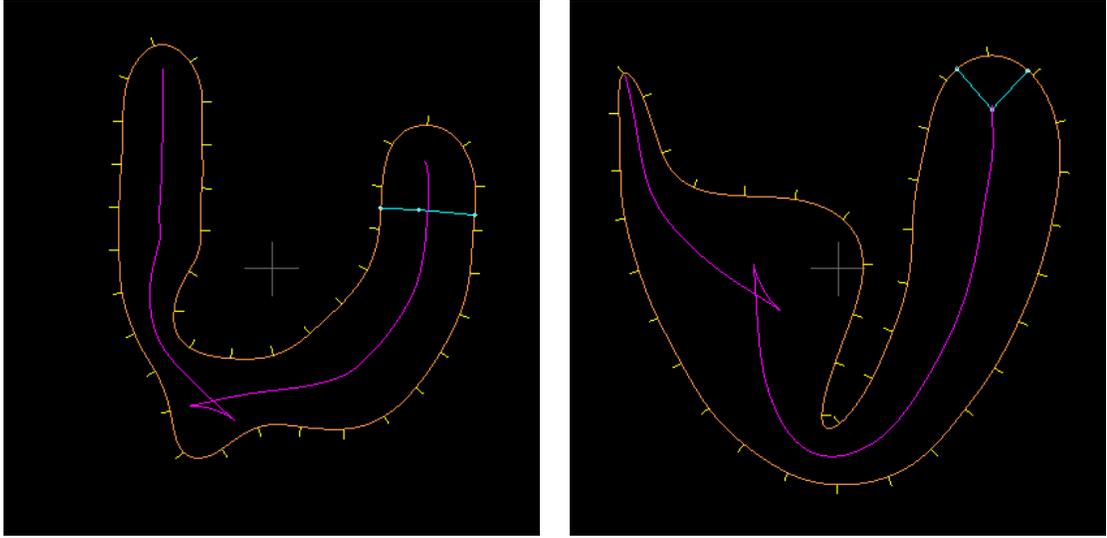


Figure 3.7: [Paul Yushkevich] Boundaries with a *non-Blum* medial axis. When  $|\nabla r|$  is too large or other “Blumness conditions” are violated, a continuous medial axis will exhibit catastrophic behavior such as cusps and swallowtail events.

$\mathbf{v}_2 = \mathbf{p}_{ij} + \vec{v}_2$  can be established as

$$\mathbf{v}_1(u, v, t) = \mathbf{v}_1(i, j, 1) \quad (3.2)$$

$$\mathbf{v}_2(u, v, t) = \mathbf{v}_2(i, j, -1) \quad (3.3)$$

where  $1 \leq u \leq N$  and  $1 \leq v \leq M$  and (as will be explained)  $-1 \leq t \leq 1$ . Thus, the involute position is parameterized by the medial atom’s mesh indices and with  $t = 1$  or  $-1$  indicating the top or bottom involute. For end atoms, the boundary position  $\mathbf{b} = \mathbf{p}_{ij} + r\eta\vec{b}$  is parameterized as well by

$$\mathbf{b}(u, v, t) = \mathbf{b}(i, j, 0). \quad (3.4)$$

These relationships are shown in Fig. 3.8. Once the medial coordinates have been assigned to the boundary positions, they are interpolated over the boundary when it itself is interpolated (based on the involute positions and normals, as will be described in Ch 4). When a subdivision algorithm is used for the boundary interpolation, a simple midpoint subdivision is used for the corresponding  $(u, v, t)$ -coordinates. Thus, top and bottom surfaces retain their  $t = 1$  or  $t = -1$  character, and coordinates at known boundary involutes do not change. Along the crest regions, the midpoint subdivision interpolates the  $t$  value between 1 and  $-1$  and interpolates the  $u$  or  $v$

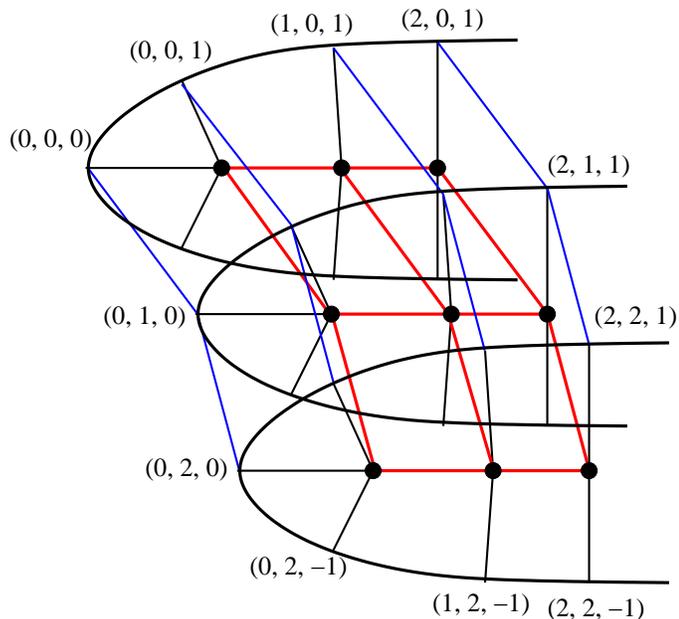


Figure 3.8: Parametric correspondence between medial atoms and boundary involutes.

value while holding constant the other along that side of the crest.

Often in subdivision methods, texture coordinates will be subdivided using the same splitting and perturbation scheme as for the vertices, a technique developed by DeRose [43] at Pixar. For m-reps, however, this would result in “coordinate creep” near the crest, with top or bottom surface points getting values of  $t < 1$  and  $> -1$ , and it would also cause medial coordinates to shift from known values at sampled medial involute positions being interpolated.

Approximate medial correspondence can be established, thus, between interpolated boundary positions and interpolated medial positions, without dependence on an exact medial interpolation—which is difficult and will be discussed below in Sec. 3.3—nor upon on exact medial correspondence. These figural coordinates are therefore well-defined for non-Blum situations, since they depend only on the initial (known) medial positions and involutes.

The  $(u, v, t)$ -coordinate system parameterizes the medially implied surface of the object in a way that is invariant under similarity transformation. By adding an addition parameter  $\tau$  representing  $r$ -proportional distance from the boundary, it is possible to extend this parameterization to the space inside and outside of the object,

subject to certain restrictions. The point  $\mathbf{y}(u, v, t, \tau)$  is thus defined by

$$\mathbf{y}(u, v, t, \tau) = \mathcal{B}(u, v, t) + \tau r_{\mathcal{B}} \vec{n}_{\mathcal{B}} \quad (3.5)$$

where  $\mathcal{B}(u, v, t)$  is the boundary position corresponding to the given  $(u, v, t)$  values,  $r_{\mathcal{B}}$  and  $\vec{n}_{\mathcal{B}}$  are the interpolated radius and surface normal at the boundary point, and  $-1 \leq \tau \leq \alpha$  gives the distance from the boundary position  $\mathcal{B}(u, v, t)$  in units of  $r_{\mathcal{B}}$ —thus,  $\mathbf{y}(u, v, t, -1)$  gives the medial position and  $\mathbf{y}(u, v, t, 0)$  the boundary point  $\mathcal{B}(u, v, t)$  itself. The upper limit  $\alpha$  on the value of  $\tau$  is dependent on the surface geometry. In a concave or saddle-shaped region, the  $\tau$  value must be less than the minimum radius of positive curvature for the surface at  $\mathcal{B}(u, v, t)$  to make it a local nearpoint to the  $\mathbf{y}(u, v, t, \tau)$  point in space. There need not be similar restrictions on values of  $\tau < 0$  related to curvature in convex regions, because, given (a)  $\tau = -1$  corresponds to the medial point itself, (b) the internal radii of curvature are minimal at the crests where their value is  $\geq r$ , and (c)  $\tau \geq -1$ , it is therefore known that  $\|\mathcal{B}(u, v, t) - \mathbf{y}(u, v, t, \tau)\| \leq r$  for  $\tau < 0$ . More accurately,  $\tau$  is in units of  $\eta r$ , with  $\eta \geq 1$  and  $> 1$  only for elongated crest regions. The same argument holds.

### 3.1.6 Restrictions on boundary curvature of m-reps

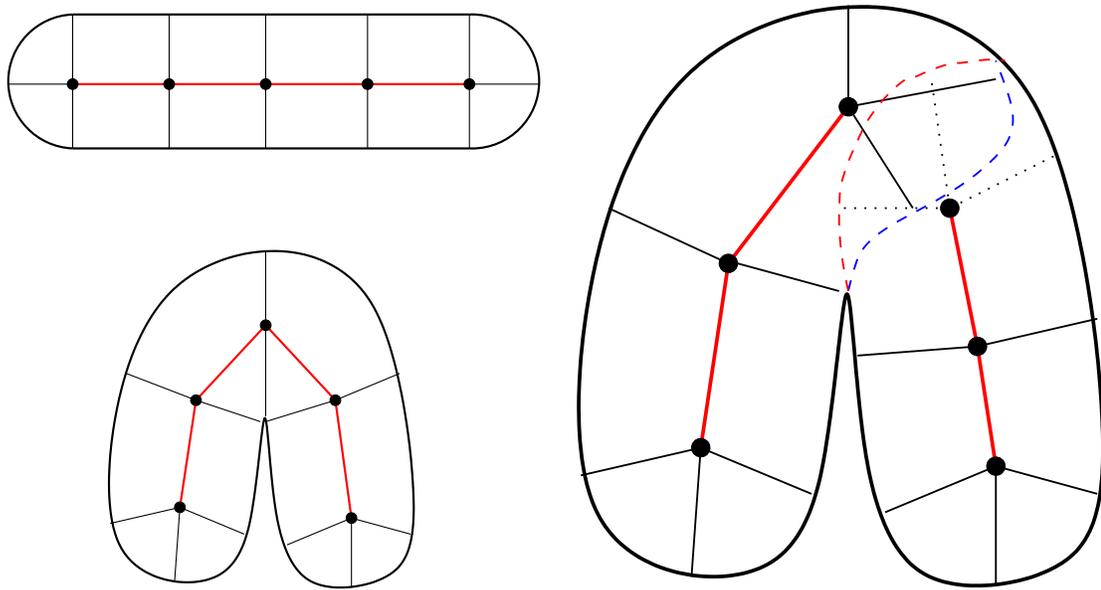
It is a postulate of this work that object shape must be captured within width-proportional tolerances; the rationale for this was presented in previous sections. Therefore, surface deformation must be within width-proportional tolerances. In addition, perturbation in the normal direction at a boundary point must be within the (signed) radius of curvature at a point to avoid surface self-intersections in the local neighborhood of the surface point.<sup>3</sup> Thus, boundary curvature must be width-proportional as well, or else there must be an additional, lower limit placed on allowable deviation. An example, of a fat sausage being bent, is shown in Fig. 3.9(a). A real object probably would show buckling and pleating at the point of maximum curvature; this can be (a) avoided, by constraining the allowed bending, or (b) modeled realistically using volume-preserving deformation methods, such as Gentaro Hirota’s free-form deformation techniques [80].<sup>4</sup>

A more m-rep-ish solution would be to model the bent object by two blended fig-

---

<sup>3</sup>This does not prevent *global* self-intersections, which require other methods, e.g., see Capell [24].

<sup>4</sup>Work on finite element modeling using m-reps for physically based deformation was the focus of dissertation work by Jessica Crouch at UNC-Chapel Hill. See the discussion in Sec. 6.3.



(a) *Bending disproportionate to width*      (b) *Avoidance by two-figure model*

Figure 3.9: An m-rep with with a bend disproportionate to object width, beyond its curvature tolerance. The image on the right shows a two-figure m-rep which avoids this problem.

ures, as in Fig. 3.9(b). This might be necessary as well for modeling of objects which are no longer *Blum*, i.e., have boundaries that cannot be produced by a smooth, unbranching medial surface, as already seen in Fig. 3.7. Such objects have medial curves with swallowtail cusps and other “bad” behavior; alternately, an object may have a smooth, continuous axis but cusps or folds on the boundary, due to inadequate constraints on the  $r$ -function along the medial curve or surface, as in Fig. 3.10 on the following page. Tests for *Blumness* have been explored by Fletcher and Yushkevich in their work on continuous medial axes as well as in the theoretical results of Damon [41] and of Giblin [70]. It is a strength of m-reps, however, that single-figure models can often represent such non-Blum figures, due to the boundary tolerances and displacements, as will be shown in later chapters.

## 3.2 Figural hierarchies

Because m-reps do not use a branching skeletal structure, a boundary displacement exceeding the allowable tolerance is represented as an attached subfigure. This allows the logical division of an object’s shape into a quasi-hierarchy made up of several possible types of figural relationships:

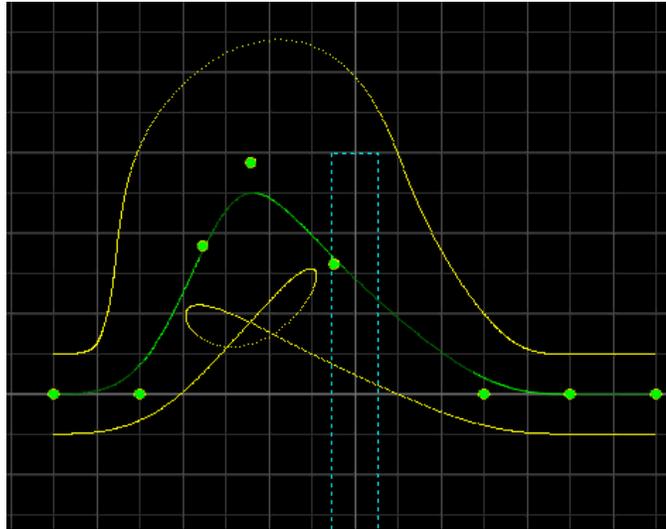


Figure 3.10: A continuous m-rep with a knotted boundary due to inadequate constraints on the medial  $r$ -function. [Thanks to Paul Yushkevich for use of his spline-based cm-rep prototype code for this image.]

- figure-subfigure relationships, where the parent figure is the main body of the object;
- co-figure relationships, where no unique parent-child relationship is apparent;
- figure-self relationships, where, e.g., the “tail” of a figure may be attached as a protrusion from part of the same figure;
- figure-*neighbor-figure* relationships, where associated figures may be unattached to the main figure (and may be internal or external) but use the figural hierarchy to specify the geometric relationship between figure and neighbor just as for attached figures.

Thus, one can construct a directed graph (not necessarily a tree) describing figural relationships within a multifigure object. Fig. 3.11 illustrates some of the possible inter- and intra-figural relationships.

A subfigure is attached at the boundary of a parent figure along locations specified by their  $(u, v, t)$ -medially based surface coordinates—thus differing from traditional skeletal modeling methods, which employ a connected, branching skeleton. This is in accord with theoretical results arising from the study of multiscale medial axes; in generic cases, a multiscale medial axis or *core* does not branch; rather, protrusions and indentations have separate axes at different levels of scale (see Damon [37]). Fig. 3.12

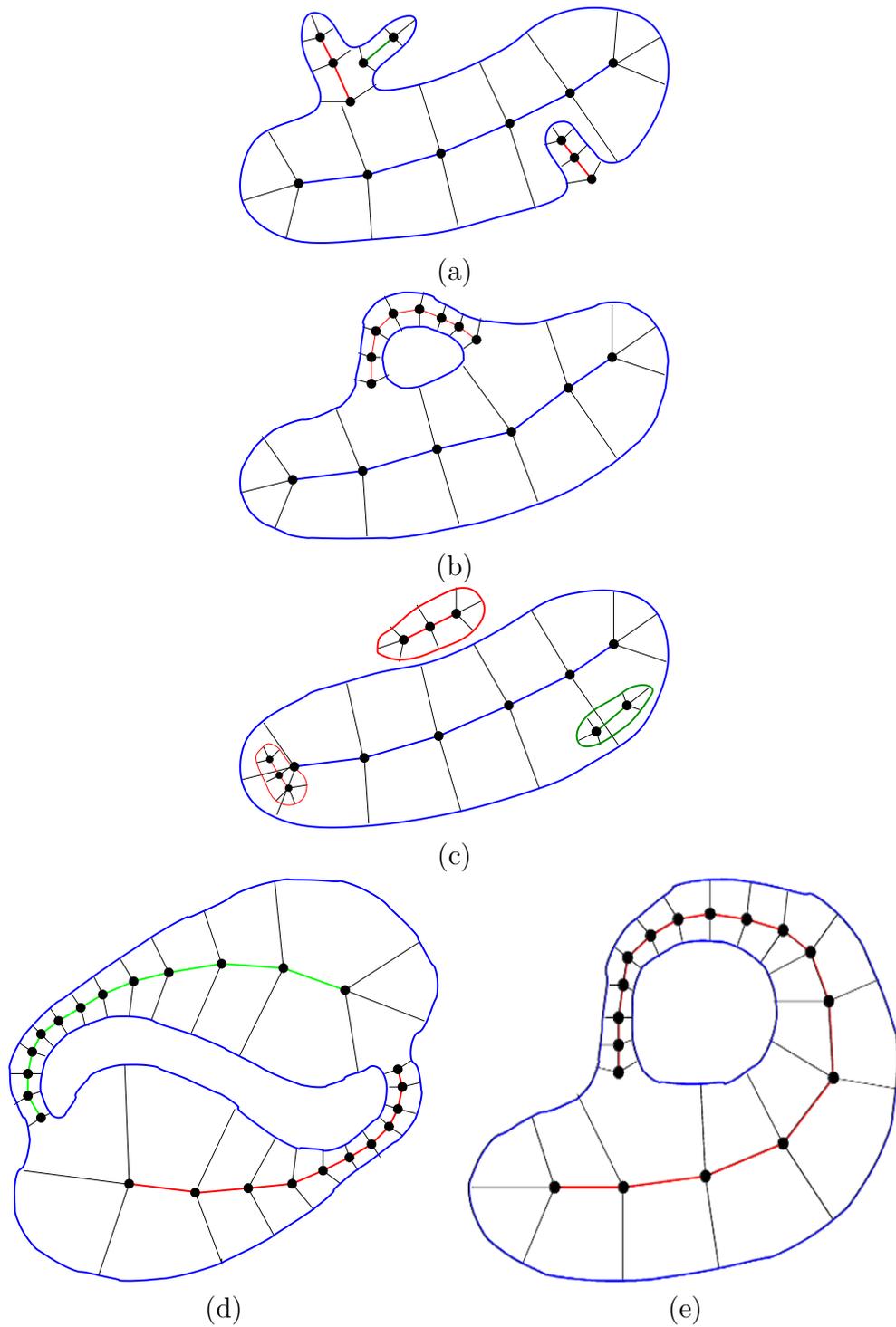


Figure 3.11: Inter- and intra-figural relationships. Figure (a) shows parent-child relationships; Figure (b) shows a child-figure with multiple attachments to the parent; Figure (c) shows figure-neighbor-figure relationships; Figure (d) shows a co-figure relationship; Figure (e) shows a figure-self relationship.

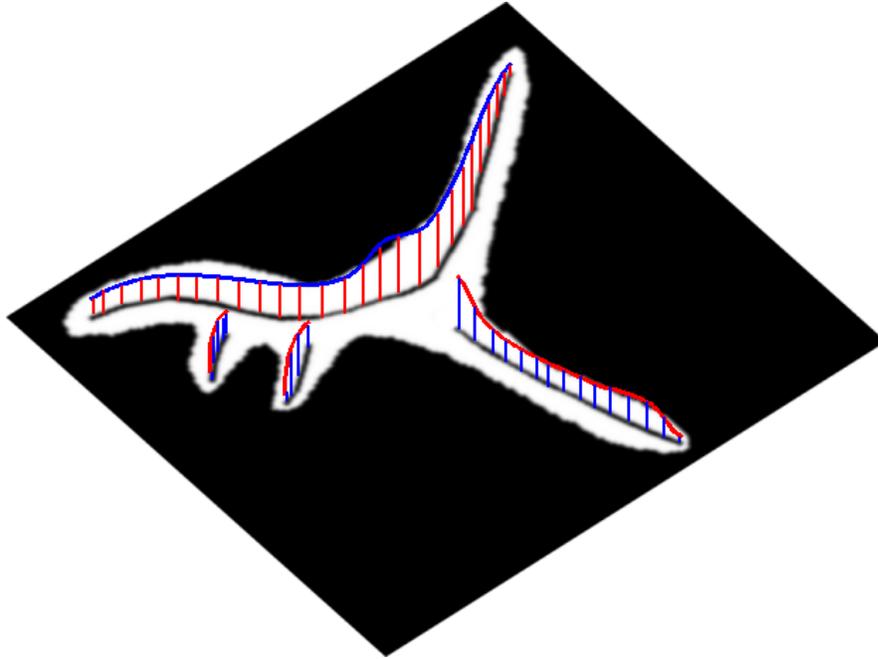


Figure 3.12: An illustration of the non-branching of multiscale medial axes in scale-space for an object in a  $2D$  image. The space curves represent the medial axis path in a  $3D$  scale-space, where height above the image approximates  $r$  at that point on the axis. The medial axes of the subfigures increase in scale as they join the main figure, then *break*—cease being ridges of maximal medialness for the object in scale-space—rather than connecting to the parent axis.

illustrates this idea. The subfigure attachment positions are thus subject to the figural boundary tolerances of the parent exactly as are other boundary attributes. Under deformation, the subfigure attachment moves in  $(u, v, t)$  and the subfigure itself is deformed in accord with the deformation of the parent-figure in the neighborhood of  $(u, v, t)$ . Figure-subfigure deformation will be discussed in Ch. 5.

Attached subfigures may form either protrusions or indentations, allowing approximate Boolean operations to be performed using m-rep figures as primitives. Fig. 3.13 illustrates this for a  $2D$  m-rep, with a main figure and two attached subfigures, one of which is an indentation figure and the other of which has a protrusion subfigure of its own. Fig. 3.14 shows a  $3D$  m-rep and subfigure as generated by Pablo. Pablo initially used implicit surfaces to compute blended boundaries between figures and subfigures and could not fit blended surfaces to indentation figures; this was an implementation issue, not a theoretical restriction, since implicit construction can be either additive or subtractive. The subdivision-surface-remeshing now used for subfigure attachment allows both additive and subtractive subfigures. Object synthesis can thus be based

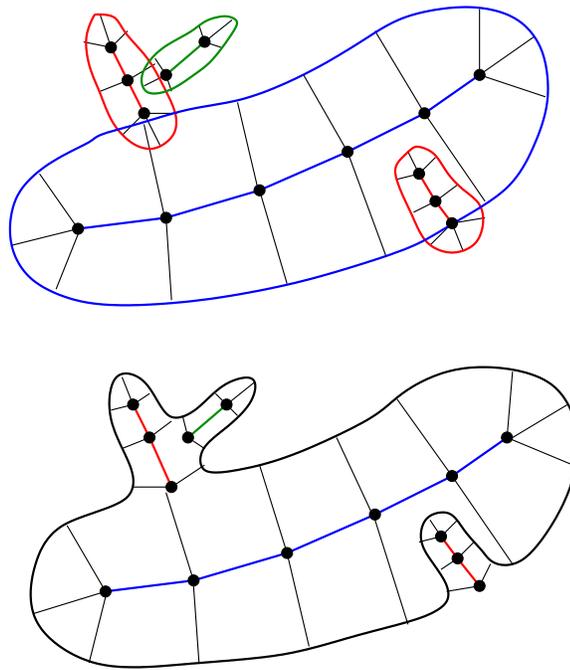
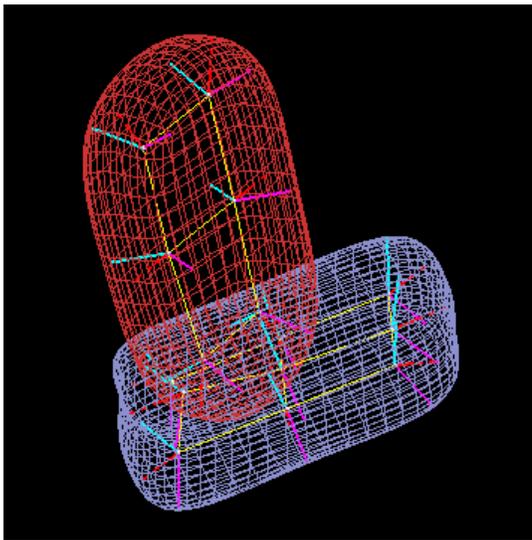


Figure 3.13: A 2D m-rep with 2 attached subfigures—a protrusion and an indentation figure. The protrusion subfigure has a subfigure of its own. Attachments of the subfigures are in the medially implied boundary coordinates of their respective parent.



*Wireframe*



*Blended*

Figure 3.14: A 3D m-rep with figure and subfigure generated by Pablo. This example uses implicit surface blending; current Pablo code uses subdivision surface remeshing for figure-subfigure joins (see examples in Ch. 5).

on sequential attaching of protrusions and indentations to an object, as approximate Boolean addition and subtraction operations allowing a CSG approach. Chapter 5 will discuss subfigure attachment by subdivision surface remeshing, and Chapter 7 will discuss its use in modeling interfaces.

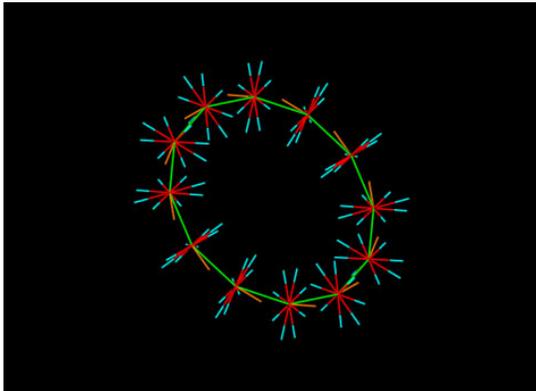
### 3.2.1 Representational polymorphism

Because m-reps take a CSG-style approach to shape modeling, using protrusion and indentation subfigures, they have an inherent representational polymorphism. While an object will have a unique Blum medial skeletonization, there can be many equivalent m-rep combinations producing the same 3D object. In fact, the tolerance-based nature of m-reps requires that there be continuous families of m-reps, with mean and statistical deviations, describing any particular shape. Different families may describe the same complex shape, and representation can and should be based on desired modes of deformation. For example, a torus can be seen as either a tube connected end-to-end or an ovoid with a hole through; thus, an m-rep of a torus can be represented as either a toroidal mesh of medial atoms or as a slab-figure with a piercing indentation subfigure. See Fig. 3.15.

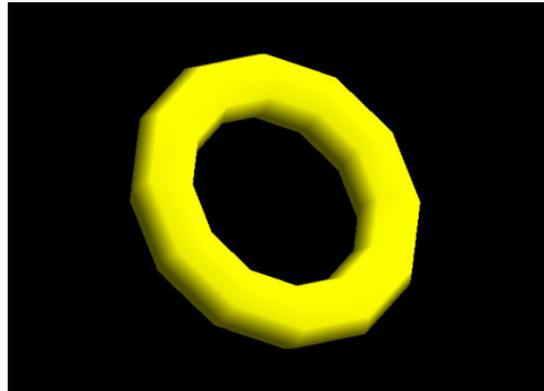
In designing m-reps to be used as deformable models—e.g., for image-based segmentation applications or statistically varied object instancing—the designer should therefore choose a structure which reflects the desired modes of variation most directly. In the case of a torus, if it seen as a ring to be twisted, stretched, or bent, then a single-figure, toroidally linked structure is most appropriate. If it is seen as a solid with a hole which may expand or contract, shift in location, or have various surface characteristics (rifling or a screw-like displacement texture) as perhaps a machined part, then a blended figure-subfigure combination is most appropriate.

## 3.3 Interpolation of medial atoms

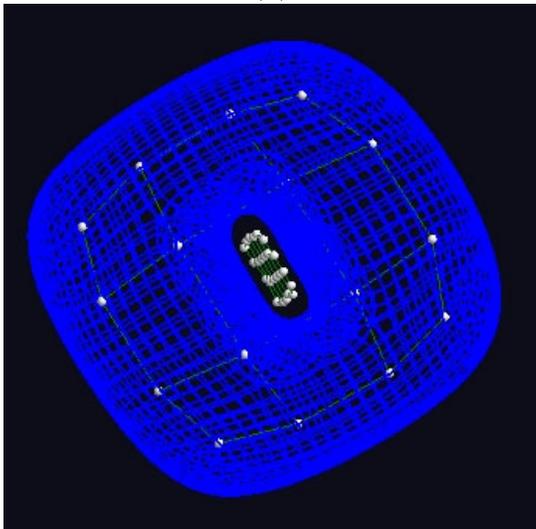
Given a sampled medial axis, there are numerous methods to construct a boundary for it, based on the boundary involutes, surface normals and other information provided by the medial atoms. These include simple boundary tessellations based on involutes, implicit surface methods, fitting of spline-based or subdivision surfaces to the boundary, or adaptive refinement of boundary meshes based on interpolative-refinement of the medial mesh to generate a finer mesh of involutes. The m-rep-to-b-rep trans-



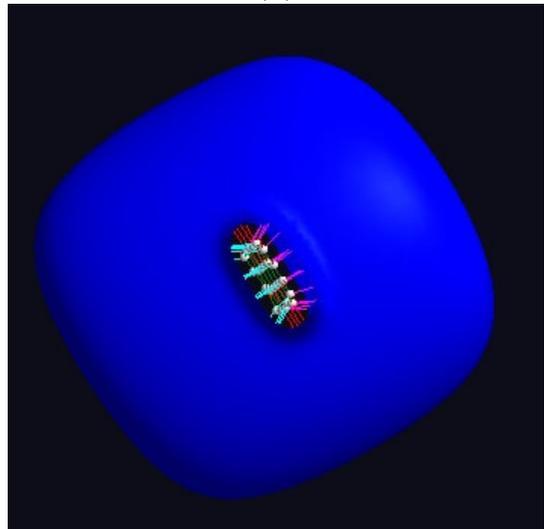
(a)



(b)



(c)



(d)

Figure 3.15: A 3D object with a hole, achieved by a 1- or 2-figure m-rep. Images (a) and (b) show a single-figure torus. Images (c) and (d) show a torus made by a figure with piercing subfigure.

formation will be discussed in the next chapter, but there is a preliminary question which can be asked here: how can we describe the medial relationships of the created boundary surface in terms of the generating medial mesh? That is, how can the implied medial surface be interpolated based on the coarse medial sampling?

I have explored three approaches to this problem, two based on a Blum point-of-view and one of which takes a tolerance-based, MMA approach. These are

- fitting a spline or interpolating subdivision surface to the medial grid and interpolating the entire  $\mathcal{M}_{ij} = \{\mathbf{p}_{ij}, F_{ij}, r_{ij}, \theta_{ij}\}$   $2D$  medial mesh for either arbitrary  $(u, v)$  parameterized positions on  $\mathcal{M}$  (for a spline fit) or for discrete medial positions on a subdivided medial grid;
- fitting a boundary and then computing (using a root-finding technique, perhaps) interpolating medial positions;
- fitting a boundary and then using the  $\mathcal{B}(u, v, 1) : \mathcal{B}(u, v, -1)$  correspondence to imply perturbed medial involutes on the boundary and computing a perturbed medial atom with these as involutes.

Independent work (and some collaboratively with me) on medial interpolation has been done by Fletcher [54], Yushkevich [164, 163], and Crouch [33]. Related theoretical results have been produced by Fletcher, Yushkevich, Damon [38], and Giblin [70]. I will discuss the above three techniques in the three subsections that follow.

### 3.3.1 Interpolating a continuous Blum axis from a sampled mesh

I explored two approaches for interpolating medial sheets: one, based on fitting Bezier patches to each quad in the initial medial mesh, interpolating  $r$  as well as  $(x, y, z)$  locations for the surface; the other, an iterative, interpolating subdivision of a medial quadmesh to a desired density to create boundary involute meshes for surface definition and rendering. This latter technique was used prior to the use of subdivision surfaces for boundary fitting.

#### Bicubic medial interpolation

It might seem that fitting a mesh of bicubic Bezier patches to the mesh of medial atoms would be straightforward; this was the first method I developed. For a mesh

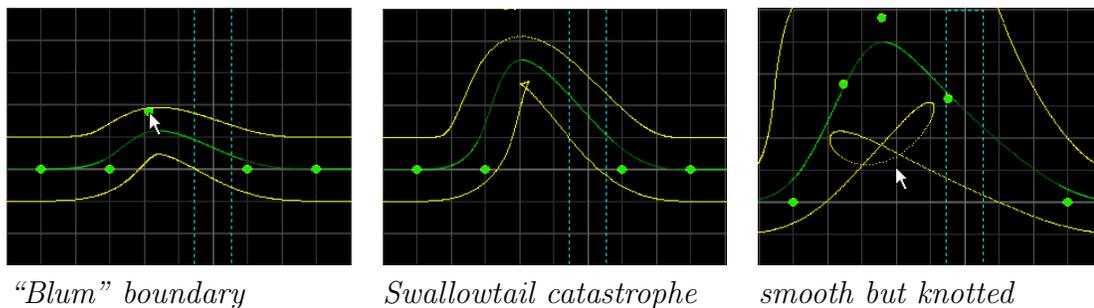


Figure 3.16:  $2D$  M-reps which fail to be “Blum” due to an unconstrained  $r$ -field. Left: a Blum boundary. Middle: a boundary with a swallowtail catastrophe, cusps of infinite curvature. Right, a boundary with a knot-shaped fold but finite curvature everywhere. (This code, by Yushkevich, uses continuous b-splines for the medial positions and radii. Similar problems arise using Bezier splines and in  $3D$ .)

$\mathcal{M}_{ij} = \{\mathbf{p}_{ij}, F_{ij}, r_{ij}, \theta_{ij}\}$ , bicubic Bezier patches  $\mathbf{p}_{ij}(u, v)$  are created for each quad of medial positions  $\{\mathbf{p}_{i,j}, \mathbf{p}_{i,j+1}, \mathbf{p}_{i+1,j+1}, \mathbf{p}_{i+1,j}\}$ , with the control points chosen so as to interpolate the positions and the medial surface normal  $\vec{n}_{ij}$  (known from the frame  $F_{ij}$ ) at each medial atom and to get  $\mathcal{C}^1$  continuity at patch boundaries. At the same time a bicubic interpolation  $r_{ij}(u, v)$  is done based on the  $r_{ij}$  and their derivative values at the atom locations. The result is a piecewise  $\mathcal{C}^2$  function,  $\mathcal{C}^1$  at the patch boundaries, giving  $\{\mathbf{p}_{ij}(u, v), r_{ij}(u, v)\}$ . Because  $r_{ij}(u, v)$  is at least  $\mathcal{C}^1$  everywhere on the interpolating surface, it is possible to analytically compute  $\nabla r_{ij}(u, v)$  and therefore, with the analytic  $\vec{n}(u, v)$  derivable from  $\mathbf{p}_{ij}(u, v)$ , to compute both  $F_{ij}(u, v)$  and  $\theta_{ij}(u, v)$  for the interpolated surface, giving a full medial atom

$$\mathcal{M}_{ij}(u, v) = \{\mathbf{p}_{ij}(u, v), F_{ij}(u, v), r_{ij}(u, v), \theta_{ij}(u, v)\}$$

for  $u, v \in [0, 1]$  on each patch.

Difficulties arise when one forgets to pull back the  $\nabla r(u, v)$  field to the parameter space for the surface patch, since the gradient—which defines  $-\vec{b} \cos \theta$ —is with respect to arc-length on the surface (see Fletcher [57]). A more serious difficulty results from lack of proper constraints on the behavior of the interpolating  $r(u, v)$  function with respect to the curvature of the medial surface to ensure that the boundary remains “Blum”. Without such constraints on  $r(u, v)$  and its derivatives, surfaces may be generated which fold or self-intersect, as illustrated in  $2D$  in Fig. 3.16. Fletcher and Yushkevich explored this problem, extending the theoretical work of Nackman [107], and derived conditions for testing an interpolating medial surface for “Blumness”;

James Damon derived mathematical results for these same (or equivalent) conditions based on the properties of curves and surfaces with generalized offset fields, of which Blum axes are a special case [39, 40, 41]. The tests involve operations such as checking the sign of the Jacobian of the medial-to-boundary function.

Paul Yushkevich dealt with this problem (and some others) by shifting to a *cm-rep*, a continuous, spline-based medial representation rather than a sampled m-rep, where objects were designed directly by modifying positions and radius values for b-spline control points for the medial surface, thus giving more control over the properties of the spline-based functions. He was able to create multifigure models in 2D and single-figure 3D models with his cm-reps. Given their explicit interpolating representation, it was more straightforward to implement Blumness criteria for the generated boundaries.

### Subdivision-based medial interpolation

My work on the above method, while predating that of Fletcher, Yushkevich, Crouch, et al., was stalled first by a mistake in the pullback function on  $\nabla r$  and then by the general lack of understanding at the time of the necessary constraints to preserve Blumness. Instead, the method I explored and used initially in Pablo was an interpolating medial subdivision based on separately interpolating  $\mathbf{p}$ ,  $F$ ,  $r$ , and  $\theta$  for new edge and face medial atoms for a subdivision of an initial  $M \times N$  diatom mesh. The method used various heuristics to independently interpolate the frame, width, and object angle, while sticking with a Bezier interpolation of the position  $\mathbf{p}$ . This technique met with limited success—the interpolated boundary involute positions were only a very crude approximation to what seemed intuitively to be the correct interpolated boundary. One difficulty was that the medial frames could not be naïvely interpolated according to (for instance) a lerp or *slerp* formula. A standard quaternion averaging of two orientations  $q_1$  and  $q_2$  would be  $Lerp(q_1, q_2, h) = (1 - h)q_1 + hq_2$  for  $h \in [0, 1]$ , or  $Slerp(q_1, q_2, \alpha) = q_1(q_1^{-1}q_2)^\alpha$  for  $\alpha$  in the same  $[0, 1]$  range.<sup>5</sup> Such quaternion methods fail to keep the  $\{\vec{b}, \vec{b}^\perp, \vec{n}\}$  vectors oriented with the  $\{\vec{b}, \vec{b}^\perp\}$  vectors in the medial tangent space and the  $\vec{n}$  vector perpendicular to it. Heuristics were

---

<sup>5</sup>Numerically, this is usually computed as

$$Slerp(q_1, q_2, \alpha) = \frac{\sin(1 - \alpha)\theta}{\sin \theta} q_1 + \frac{\sin \alpha\theta}{\sin \theta} q_2$$

where  $\alpha \in [0, 1]$  and  $\cos \theta = q_1 \cdot q_2$ , as given in the canonical Shoemake paper [135].

necessary instead for rotating the frame about the medial normal, with special cases required for crest atoms. The subdivision method was used, rather than continuous interpolation, in an attempt to smooth over problems with these techniques and only subdivide as finely as necessary.

This method was discarded as *ad hoc* and overly complex once alternative methods were made possible by the switch to a subdivision-based explicit boundary. In terms of medial-based approaches, Paul Yushkevich’s b-spline based cm-reps are the correct solution—since the medial meshes used here are regular quad-grids, a Catmull-Clark subdivision of them is equivalent to a b-spline fit, and Yushkevich’s methods correctly interpolate  $r$  and its gradient to give the correct medial frames. Another approach to this in the future may be provided by the Lie algebra methods for medial-atom interpolation developed by Fletcher et al. [55]; work on correct medial interpolation using these methods has been undertaken by Kerckhove [91].

### 3.3.2 Interpolating a Blum axis from an m-rep-based b-rep

As already stated, there is a tremendous body of literature on skeletonization and the extraction of medial axes from  $2D$  and  $3D$  b-reps. The problem under consideration here is different: given a b-rep generated by an underlying m-rep, how can the medial structure defined by this object be interpolated for medial atoms away from those sampled ones. An application for this from  $2D$  image analysis was the desire to resample an m-rep to maintain an equal distance (in an  $r$ -proportional sense) between each atom in a  $2D$  chain to establish a homology across a population of images for statistical studies (Yushkevich [165]). In this way, m-reps could provide a consistent correspondence using atoms as shape features, invariant under limited deformations. The key insight for the interpolation was that, given a boundary representation based on a spline boundary fit to an m-rep skeleton, one can attempt to find interpolating medial positions based on a root-finding approach taken on offsets from the boundary surfaces on opposing sides of an object.

The  $2D$  method devised by Yushkevich had three steps:

1. Hermite-interpolating splines are fit to the boundary involutes and normals, giving a  $\mathcal{C}^2$  continuous boundary;
2. a point is selected on the boundary;
3. an equation is solved giving as its root the corresponding medial involute on a

spline on the opposite side, if it exists.

These are not the previously mentioned cm-reps—they represent a boundary-to-medial approach developed prior to his medial b-spline methods. This technique was possible due to the ability to solve for the medial involute in closed form given the bounding cubic spline curves.<sup>6</sup> This method had problems with non-Blumness, similar to the case of the spline-based medial curves, but producing instead cusps and self-intersections on the medial curve derived from the spline-based b-rep. Significantly, Yushkevich tagged medial regions where the object failed to be Blum, and this raised awareness within MIDAG that led to the practical and theoretical ideas for constraints to ensure Blumness.

Yushkevich applied his method to simple 3D m-reps with bicubic spline boundaries as well. My own work in 3D, which will be dealt with in detail in Ch. 4.3, was based on trying to achieve a similar result for an m-rep with a subdivision surface boundary. No closed-form equations for these boundaries are available, unless one fits b-splines to the subdivision boundaries and performs Stam-style interpolation in extraordinary regions [137]. Another method is possible, however. Given point  $\mathcal{B}(u, v, t)$  on the boundary, an opposing involute  $\mathcal{B}(u', v', t')$  on the opposite side can be solved by using a numerical root-finding technique. The paired involutes thus define an approximate medial atom—depending on the subdivision level of the mesh—which approaches a correct Blum medial atom for the limit surface as the mesh is refined.

### 3.3.3 Interpolating a multiscale axis from $(u, v, t)$ correspondence

A major point of m-rep based medial modeling is to avoid the rigid overprecision and intolerance of Blum-based methods. For interpolation of new medial atoms based on a particular m-rep, the tolerance-based nature of m-rep modeling actually means that there is no single *correct* interpolating atom, but rather a whole family of possible interpolants. In this subsection, I will describe a simple method for medial interpolation in a tolerance-based, multiscale framework; more advanced methods based on Lie algebras and other methods will be discussed in Ch. 6.

---

<sup>6</sup>In fact, Yushkevich originally used Pythagorean hodograph curves for his boundaries, 5<sup>th</sup>-order curves which act like cubics with additional properties of arc-length parameterization and rational offset curves. These were studied by Farouki et al. [52, 51] in the mid-1990s; they do not, alas, generalize to 2D surfaces.

Parametric correspondences on a m-rep are established between boundary points by the involute locations of the sampled medial atoms. This allows *approximately medial* atoms to be found that associate boundary positions that are approximate medial involutes of one another. Given a  $(u, v, t)$  parameterization of the boundary as discussed above in Sec. 3.1.5, a medial atom can be fit to the boundary points  $\mathcal{B}(u, v, t)$  and its associated  $\mathcal{B}(u, v, -t)$ . Because of the somewhat arbitrary nature of the boundary parameterization (dependent on midpoint subdivision rules in the current implementation), it is seldom that these are corresponding medial involutes in a Blum sense, nor is it necessary that they should be, given the boundary tolerances already built into the model. It is enough to establish a metric for the deviation of the relationship from involution and to show that the deviation is acceptably small in most cases.

To create an approximately medial atom for a given pair of corresponding boundary points, a technique was developed that projects from the boundary positions in their normal directions and finds an averaged medial point. The algorithm has three steps:

1. project associated points  $P_a$  and  $P_b$  along their respective normals  $\vec{n}_a$  and  $\vec{n}_B$  inward by their respective  $r_a$  and  $r_b$  values as interpolated on the boundary;
2. connect these projected positions by a line segment;
3. project the midpoint of this segment to a point  $\tilde{\mathbf{p}}$  on an approximated medial plane;
4. taking this as the approximate medial position, create vectors  $\vec{v}_1$  and  $\vec{v}_2$  from  $\tilde{\mathbf{p}}$  to the  $P_a$  and  $P_b$  boundary positions, respectively;
5. use  $\tilde{\mathbf{p}}$ ,  $\vec{v}_1$ , and  $\vec{v}_2$  to compute the components of a complete, approximating medial atom  $\{\tilde{\mathbf{p}}, F, r, \theta\}$ .

The medial plane used is the plane bisecting the chordal segment connecting the two boundary points. Figure 3.17 illustrates this method, and Fig. 3.18 shows a medial surface tiled using interpolated atom positions based on the  $(u, v, t)$  correspondences of the m-rep subdivision.

The  $r$ -proportional distance between inwardly displaced boundary locations serves well as a metric for the *deviation-from-medial-involution* of the parametrically associated boundary points. This will be discussed specifically in the context of m-rep

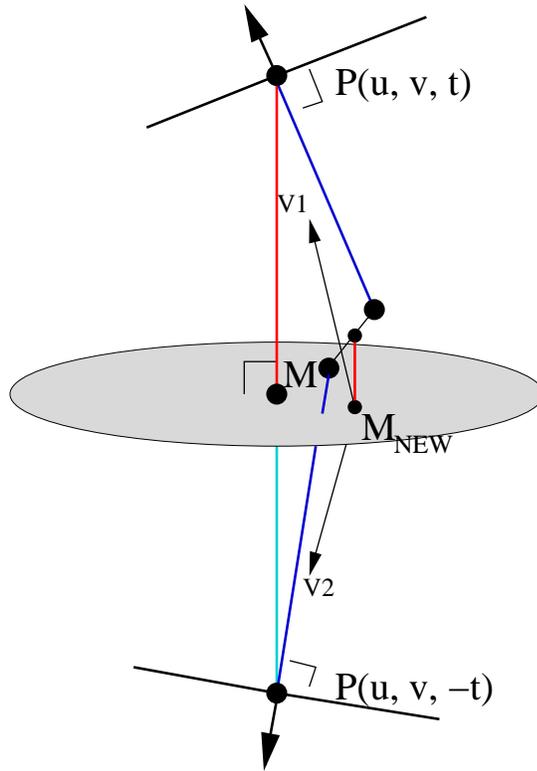


Figure 3.17: Interpolating a medial atom (with tolerance) given a boundary correspondence.

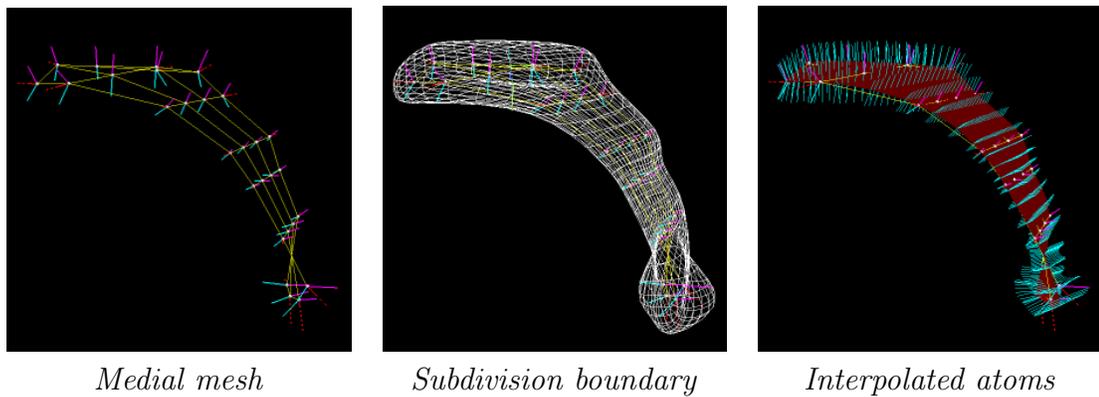


Figure 3.18: Interpolating a medial axis using boundary correspondence. New medial atoms are computed by sampling evenly in  $(u, v)$  along the  $\mathcal{B}(u, v, 1)$  and  $\mathcal{B}(u, v, -1)$  boundary surfaces and computing new medial atoms by treating the sampled boundary positions as medial involutes.

subdivision boundaries in Section 4.3. While these interpolating medial atoms can be used to resample the implicit medial surface, there are no Blumness or continuity constraints imposed on the interpolation and thus no guarantee that there is even a  $\mathcal{C}^0$  continuous medial surface being interpolated. This would create problems for Blum-based medial modeling but works well in the context of a multiscale, width-proportional representation, where medial positions themselves need not be accurate at the continuous sampling density of a Blum medial axis. The interpolated positions should thus be used for resampling only to maintain a good sampling density for the medial mesh.

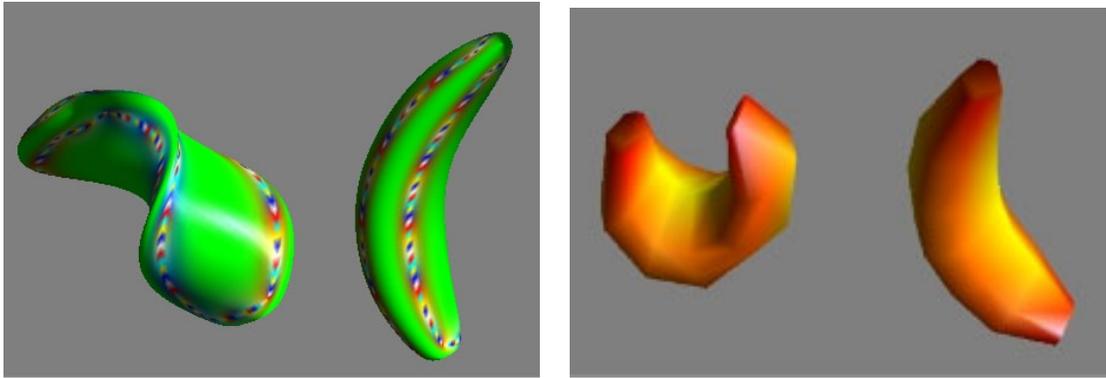
For  $\mathcal{C}^0$  interpolating boundary subdivision, the medial atoms created do not match those of the sampling medial mesh at the known integral  $(u, v, t)$  locations; given  $\mathcal{C}^1$  interpolating subdivision surfaces, however, the above algorithm gives correct medial atoms at the sampled  $(u, v, t)$  locations.

## 3.4 Topology-preserving deformation

An advantage of a skeleton-based modeling system is the ability to perform topology-preserving deformations by manipulating the underlying skeleton. In the case of m-reps, we can also produce local boundary deformations. Both of these may be based on statistical *priors* describing normal modes of variation for instances of a class of objects. Such statistical deformations have applications in image-analysis—in model-based shape segmentation—and in mechanical and other shape-modeling tasks. For example, in computer graphics, statistical descriptors can be used to select average members stochastically from a class of objects for modeling purposes. The most advanced applications of statistical deformation in image-analysis is in the previously mentioned work of Fletcher, using a Lie algebra over the medial atom components  $\{\mathbf{p}, F, r, \theta\}$  to enable principal components analysis (*PCA*) to compute the shape statistics necessary for Bayesian analysis. The discussion of deformation in this section predates Fletcher’s techniques.

### 3.4.1 Medially based object deformations as a primitive operations

While medial deformation will be discussed in more detail in Ch. 6, there is a vocabulary of primitive deformations that can be defined on a single figure given no



(a) medial-based propagated twisting      (b) medial-based propagated bending

Figure 3.19: Object deformation by medial mesh modification. These illustrate the bending and twisting operations propagated along the medial mesh; the effect is similar to an axial-based space warp, but all deformation is relative to the local medial frames.

more than the medial mesh and the medial-based coordinate frames of the individual atoms. Because each atom has its own medial frame, which can be used as a coordinate basis for the entire model, a medial deformation—a bending, twisting, rotating or scaling—can be centered at a specific mesh atom and then propagated according to any of a number of possible methods to the other atoms in the mesh, producing non-local medial deformations. The *Yaksha* program (part of the *Rakshasa* project) was designed as a prototype to test such techniques, using slice-constrained single-figure m-reps as a modeling primitive. Medial changes centered at an atom could be propagated along the mesh—e.g., to convert a bending along a row of atoms into a curling up of the medial figure, as in Figs. 3.19 and 3.20.

Other *Rakshasa* operations involved medial scaling or twisting centered at a vertex and propagating to neighboring atoms or, for a quadmesh, in one direction or the other down rows and columns. Similarly, translations can be propagated directionally along the medial mesh to produce object-elongations, using the local medial frames to center and direct the deformations.

Another means of medial deformation is the specifying of medial “joint” atoms, by their  $(u, v, t, \tau)$  positions, to serve as joints for skeletal deformation. Thus, a joint atom may be located at a position in space neither on the medial grid nor even on its implied medial axis, to serve as a pivot for medial-based bending and twisting. This is similar to the use of skeletal control points in 3D animation—virtual puppetry—but rather than having to “skin” our skeleta about joints (as per Markosian [102] and the general state-of-the-art in the animation industry), m-reps have their continuous

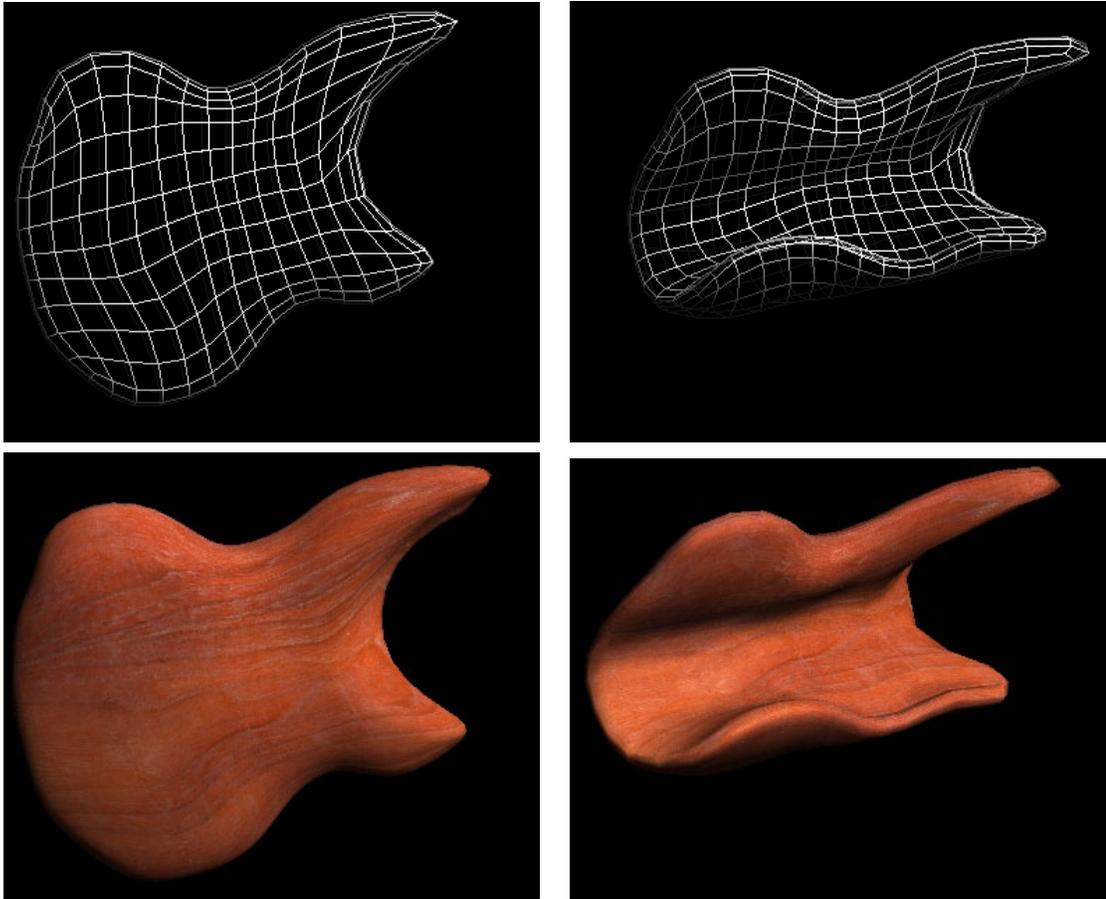


Figure 3.20: Object deformation by medial mesh modification. Another example showing the propagation of a bending operation across the medial sheet. (This instance was created manually, not using Yaksha.)

boundary structure built-in. One can place restrictions on joint movement, have “scaling joints” and other non-standard operations, and one can place procedural “textures” on joint movements as is standard in the field.

### 3.4.2 Boundary deformation

The mesh of medial atoms provides the coarse-scale shape description, leaving the fine-detail to boundary displacement and deformations. The main approach I have considered for incorporating boundary perturbation in a generic m-rep model—irrespective of the particular boundary fitting method—operates over the entire boundary by applying a scalar displacement field to either boundary patches or to subdivision vertices. This produces the fine-scale boundary definition required for m-reps to be robust and general modeling primitives. Given a medially implied boundary  $\mathcal{B}(u, v, t)$  with an associated interpolated  $r(u, v, t)$  width-function (that need only approximate the true medial width at  $\mathcal{B}(u, v, t)$ ), a displacement map  $\mathcal{D}(u, v, t)$  is created for the surface giving a perturbed boundary  $\mathcal{B}'$  such that

$$\mathcal{B}'(u, v, t) = \mathcal{B}(u, v, t) + r(u, v, t)\mathcal{D}(u, v, t)\vec{n}_{\mathcal{B}} \quad (3.6)$$

where  $\vec{n}_{\mathcal{B}}$  is the boundary normal at  $\mathcal{B}(u, v, t)$ . For modeling purposes, it may also be necessary to compute new boundary normals for the perturbed positions. Currently, new normals are computed on-the-fly by averaging neighboring vertices in the perturbed meshes. Speculatively, quaternion-textures might be used to map rotational perturbations of the normal relative to the medial-coordinate system.

For a subdivision boundary, displacements may be seen as a *MIP-map*, describing deviations in the normal direction on a per vertex basis at progressively finer subdivision levels. Considering displacements as scalar height fields, it is non-trivial to resample such displacement fields at different LODs; this difficulty arises as well in IBR research when resampling disparity maps.

While the displacement fields I consider are restricted to the  $\vec{n}_{\mathcal{B}}$  directions, it is possible to have more general, vector-valued displacements as well, or multilevel displacement strategies such as in Guskov [76]. Discussion of m-rep boundary deformation will be expanded upon in Ch. 5 on displacement subdivision surfaces, and Ch. 6 will give implementation details and considerations for both medial-based and boundary-based m-rep deformation in application.

## 3.5 Drawbacks and limitations of m-reps

This chapter concludes with a discussion of some the drawbacks and limitations of m-reps—vis-a-vis their basis in sampled medial sheets—as modeling primitives in CG and IA, both theoretically and as currently implemented. I will discuss four main topics:

- quadmeshes and their limitations as the only currently implemented m-rep mesh;
- corners, creases, and other surface discontinuities and how they can be implemented;
- the Correspondence Problem—how to establish shape homologies given the *tolerant* placement of medial atoms in an m-rep mesh describing an object;
- arguments for discrete m-reps versus cm-reps.

Each of these will be discussed briefly.

### 3.5.1 Quadmeshes and their disadvantages

Regular quadrangular meshes of medial atoms have many advantages, including simplicity of indexing and storage, and ease of converting to a quadmesh boundary for Catmull-Clark subdivision. Ch 4 discusses the latter in detail. Quad meshes are adequate for many and perhaps most modeling tasks, but the necessity for a constant number of rows and columns regardless of the needs of width-proportional sampling is at least a theoretical drawback and perhaps a practical one as well. Having to provide special behavior at quadmesh corners is another drawback. There are certainly many objects that do not naturally lend themselves to quadmesh representation—it is a tribute to the flexibility and polymorphic nature of m-reps that quadmeshes have sufficed for so much work in image-segmentation. Arbitrary trimeshes with planar connectivity are one alternative (hex-mesh simplices are another); they are trickier to store, less straightforward to index, and less trivial to generate boundary tessellations. Given such a tessellation, however, a modified Loop subdivision can substitute for Catmull-Clark, using a similar interpolating variant as that presented in the next chapter for Catmull-Clark quads. Trimeshes would provide for simpler interactive remeshing of the medial mesh to maintain a desired sampling density, and they eliminate non-planarity issues found with quad-based representations.

### 3.5.2 Corners, creases, and other surface discontinuities

As discussed above in Sec. 3.1, the medial skeleton of an m-rep alone implies an inherently blobby object with corners, creases, and other first-order discontinuities “blurred away” by the fuzziness of the representation. This has not been a problem in Pablo, where most of the anatomical objects being modelled are somewhat blobby in nature; but it can be misleading to equate the boundary has been fit to the medial skeleton for the “true” shape of the m-rep-implied object. Even if the medial atoms can be so arranged as to produce a sharp crease along a subdivision boundary, this is not the medially implied boundary but simply an accident of the boundary fitting. The medially implied boundary always includes its width-proportional tolerance, and fine surface detail must be produced at the boundary level, whether by displacement maps or subdivision vertex- and edge-freezing. It becomes a philosophical argument between Blum and non-Blum representations, wherein one argues about whether a brick is simpler than an ovoid. Creation of objects with corners and creases will be discussed in detail in Ch. 5.

### 3.5.3 The Correspondence Problem

While correspondence is not a problem for synthetic model generation, the user of m-reps in image analysis must face the problem of establishing correspondence between m-reps fitted to different members of a sampled population. Because the locations of the medial atoms may vary within the implied medial sheet and still produce a similar boundary, a naïve atom-to-atom correspondence does not suffice. In 2D, this problem was addressed by Yushkevich [165] by enforcing a maximal,  $r$ -proportional spacing of sampled atoms along the implied medial curve. This method in 3D is problematic, due to the additional degrees of freedom for medial atom displacement and due to the difficulties of computing accurate medial interpolants based on the implied boundary. Finding homologies needed for 3D m-rep-based segmentation and data analysis is the subject of ongoing research.

### 3.5.4 Discrete m-reps vs. cm-reps

Given a continuous medial representation such as the cm-reps of Yushkevich, one might ask why a discrete m-rep is required at all. This will be the topic of some debate in MIDAG over the next few years; its resolution may rest as much on convenience and the existence of a large body of legacy code as on theoretical results. On my

side of the debate, I would argue that the continuous cm-rep, as well as continuous, branching 3D medial representations by Fletcher, are Blum axis methods at heart. As such, while they have been tuned as tools for 3D shape design, and while they have really lovely mathematics behind them, they nonetheless do not avoid the intolerances and overprecision of other Blum medial methods, making them somewhat “brittle” and overly reliant on “Blunness” constraints and tricks to get medial branching to work effectively. It may be that, in practical applications, there will be little difference in their utility *vis-a-vis* discrete m-reps, but that will be because a lot of cleverness went into getting past some of their serious drawbacks.

### 3.6 Concluding remarks on sampled medial sheets

This chapter has presented the medial atom and medial mesh framework at the heart of modeling with sampled m-reps. It described a medially based coordinate system for parameterizing the boundary and the surrounding regions of space. It discussed figural hierarchies and the inherent polymorphism of shape design using m-reps, and it discussed both Blum- and non-Blum-based methods for interpolating medial locations based on a coarse sampling grid. It discussed topology-preserving deformation techniques and concluded with a brief discussion of some drawbacks and limitation to m-rep object representations.

The next chapter will present techniques for fitting boundary surfaces to an m-rep skeleton, focusing on the use of a new form of iteratively interpolating subdivision surface to produce boundaries within desired boundary tolerances. It will show how nearpoint distance tests may be done on these boundaries to give  $(u, v, t, \tau)$  coordinates for points in space near an m-rep, and it will show how such proximity tests can be used to create a *near*-Blum medial interpolation based on Phong normals of tessellated boundaries.

# Chapter 4

## M-rep Boundary Fitting by Interpolating Subdivision Surfaces

A sampled medial skeleton must be fleshed out by a surface which can carry the fine-scale geometric information. The difficulties of fitting a boundary based on medial interpolation and exact Blum medial correspondence were discussed in Section 3.3. One alternative to cm-reps is to use an implicit surface representation, treating the medial radius function as a density field with an isosurface at the medially defined boundary. Such an approach is similar to methods for rendering convolution surfaces and was explored in preliminary work by Fletcher [53] (see Fig. 4.1). Implicit representations simplify figure-subfigure blending but have several drawbacks: they still require medial resampling, as discussed previously, and they present difficulties in parameterizing the surface for medial correspondence and boundary displacement.

One alternative to a direct medial approach to boundary definition is to use the coarse medial sampling alone to derive the boundary, fitting a surface to the medially implied boundary involutes. Subdivision surfaces are ideal for these medially implied boundaries for a number of reasons:

- they allow surfaces of varying mesh connectivity and topology, requiring less attention to special cases and continuity constraints than would spline-based surface patches;
- they can interpolate boundary positions and normals for the known involute positions of the medial atoms;
- they are a multiresolution surface representation, which fits in well with the multiscale modeling paradigm of m-reps;

- they are a subject of much current research—including research on CSG-style approximate Boolean operations—and are being implemented in graphics APIs and rendering hardware.

This chapter discusses the creation and use of interpolating subdivision surfaces for m-rep boundaries. Section 4.1 describes a new iterative algorithm for interpolative subdivision and shows that it is equivalent to solving a linear system for an interpolating Catmull-Clark subdivision surface. This method is equally applicable to interpolating Loop subdivision on triangle meshes. The section also discusses methods to interpolate normals and gives an error metric (from the implied medial atoms) when normals are not interpolated. In particular, Section 4.1.3 presents a method for directly computing limit positions for irregular mesh vertices (i.e., with non-quadrangular face-neighbors), thus allowing the new interpolation technique to be used on general closed meshes. Section 4.2 describes an algorithm and associated heuristics for computing approximate nearpoints on an interpolating subdivision surface that uses Phong normals (barycentric normal interpolation across triangulated tiles) to approximate nearpoints on meshes of varying subdivision levels. In the limit, as the subdivision level increases, the nearpoint distance approaches a true normal distance from point to surface. The described technique is for Catmull-Clark surfaces but is equally amenable to Loop subdivision surfaces. The section also discusses other proximity tests such as plane-to-m-rep and m-rep-to-m-rep. Section 4.3 discusses the use of the nearpoint test for computing approximate interpolating medial atoms for the mesh, illustrating a *Phong-normal medial axis* for a tessellated surface. In the limit, as subdivision levels increase, this gives points on the Blum medial axis for the subdivision surface. This chapter concludes with some drawbacks and limitations of iteratively interpolating subdivision surfaces for m-rep boundaries.

Following these discussions, Ch. 5 will explore (a) the fine-scale modification of subdivision boundaries using displacement textures and (b) the joining of figures and subfigures represented by subdivision meshes. These are the last two ideas required to make m-reps a useful multiscale modeling primitive.

## 4.1 Iteratively interpolating subdivision surfaces

The classic methods of uniform, stationary subdivision surfaces—Doo-Sabin and Catmull-Clark—are approximating subdivision techniques, as is Loop subdivision for

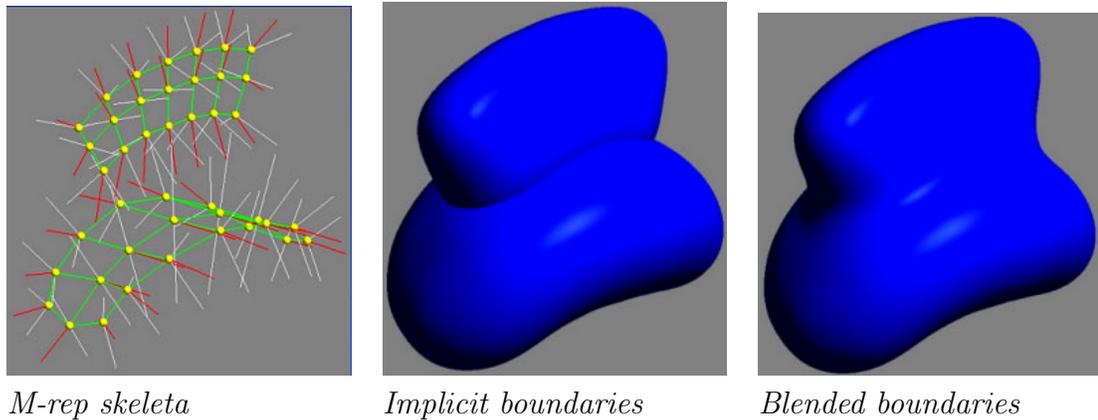


Figure 4.1: (Tom Fletcher) A figure-subfigure blend based on an implicit-function. This uses interpolated medial positions as centers of Gaussian density fields, and thus differs from the implicit blending in Pablo (Fig. 2.2) which is based on an implicit boundary function, rather than a medial one.

trimeshes.<sup>1</sup> Vertices at the coarsest level are linearly transformed at each iteration to new locations, approaching their corresponding points on the limit surface. They can be transformed immediately to these limit points using a modified subdivision matrix, and this is generally done after the surface mesh has been subdivided to adequate fineness, often adaptively based on boundary curvature approximations.

For fairing of polyhedral objects—attempting to fit a smooth surface exactly to known vertices and vertex normals—other methods have also been developed. Jorg Peters [115], for example, performed  $\mathcal{C}^1$  interpolation using piecewise-bicubic patches for mesh-fitting, with linear normal-interpolation along patch boundaries. Implicit techniques such as those of Bajaj and Ihm [5] create algebraic patches of  $\mathcal{C}^1$  continuity for closed polyhedra. Moreton and Séquin [104] employed a functional optimization approach to flesh surfaces based on point, normal, and curvature constraint sets.

Subdivision methods have risen to prominence, however, due to their conceptual simplicity, their equivalence to spline-based surfaces away from extraordinary points, and their requiring only minimal constraints on object-topology and mesh connectivity. Surface interpolation is attainable by several means. The Butterfly interpolation scheme of Dyn et al. [48] or the techniques of Zorin [166] can give  $\mathcal{C}^1$  continuity on trimeshes subject to tension constraints or other parameters. Halstead, Kass, and

---

<sup>1</sup>A reminder on nomenclature: *stationary* subdivision means that the same subdivision rules are used at each subdivision level; *uniform* subdivision means that the same set of rules are used everywhere on the mesh. Non-stationary methods may be used to adjust surface normals, as below; non-uniform methods may be used for forming cusps, edges and corners, as per DeRose [43].

DeRose [77] employed a modified Catmull-Clark technique to produce interpolating subdivision of quadmesh structures, using thin-plate and membrane energies to constrain the subdivision and interpolate both positions and normals at the desired boundary points. The advantage of Catmull-Clark surfaces and others like them is that they are almost everywhere  $\mathcal{C}^2$  and have closed-form limit positions and limit normals for vertices at any subdivision level.

### 4.1.1 Interpolating Catmull-Clark boundaries

For m-rep surface-fitting, a technique like that of Halstead et al. [77] would be satisfactory but is in fact more exact than necessary, and it pays for that exactness with its complexity and implementation details. What is needed instead is a subdivision surface that creates a limit surface with boundary positions and normals lying within the tolerance regions of the medial primitives. Such a surface should be  $\mathcal{C}^2$  as well, allowing displacements in the normal direction to be limited by the radius of curvature in concave or saddle-shaped regions. Such a technique has been developed for m-reps boundaries by means of an iterative algorithm applied to the initial, coarse mesh of involute positions, which gives approximately interpolating subdivision boundaries within the desired tolerance. Figures 4.2 and 4.3 show examples of this boundary-fitting. While the method has been applied to Catmull-Clark subdivision, it would be effective for any subdivision method where the limit masks can be simply computed for a local 1-neighborhood. While it was designed for m-rep boundaries, it can be used to interpolate any closed, two-sided polygonal mesh; Figs. 4.4, 4.5, and 4.6 show details of a polygonal mesh of a cow model interpolated by the algorithm.

The technique developed for m-rep boundaries, like that of Halstead, involves solving the linear system for an initial subdivision grid that will produce limit positions interpolating the required boundary positions. Unlike Halstead’s algorithm, however, it uses an iterative solution method, requiring only 1-neighborhoods of mesh vertices and producing successively closer approximations to the involute positions by an algorithm that is  $\mathcal{O}(m \cdot n)$ , where  $n$  is the number of vertices being interpolated and  $m$  is the number of iterations. In practice, with  $m$  small, this effectively adds a constant cost per coarse-level mesh-vertex over the cost of non-interpolated subdivision using the same mesh. Similar iterative techniques have been used in CAGD for b-spline interpolation (see Farin [50], pg. 125). The rest of this section will detail the theory and practice of *iteratively interpolating subdivision surfaces (IIS-surfaces)*.

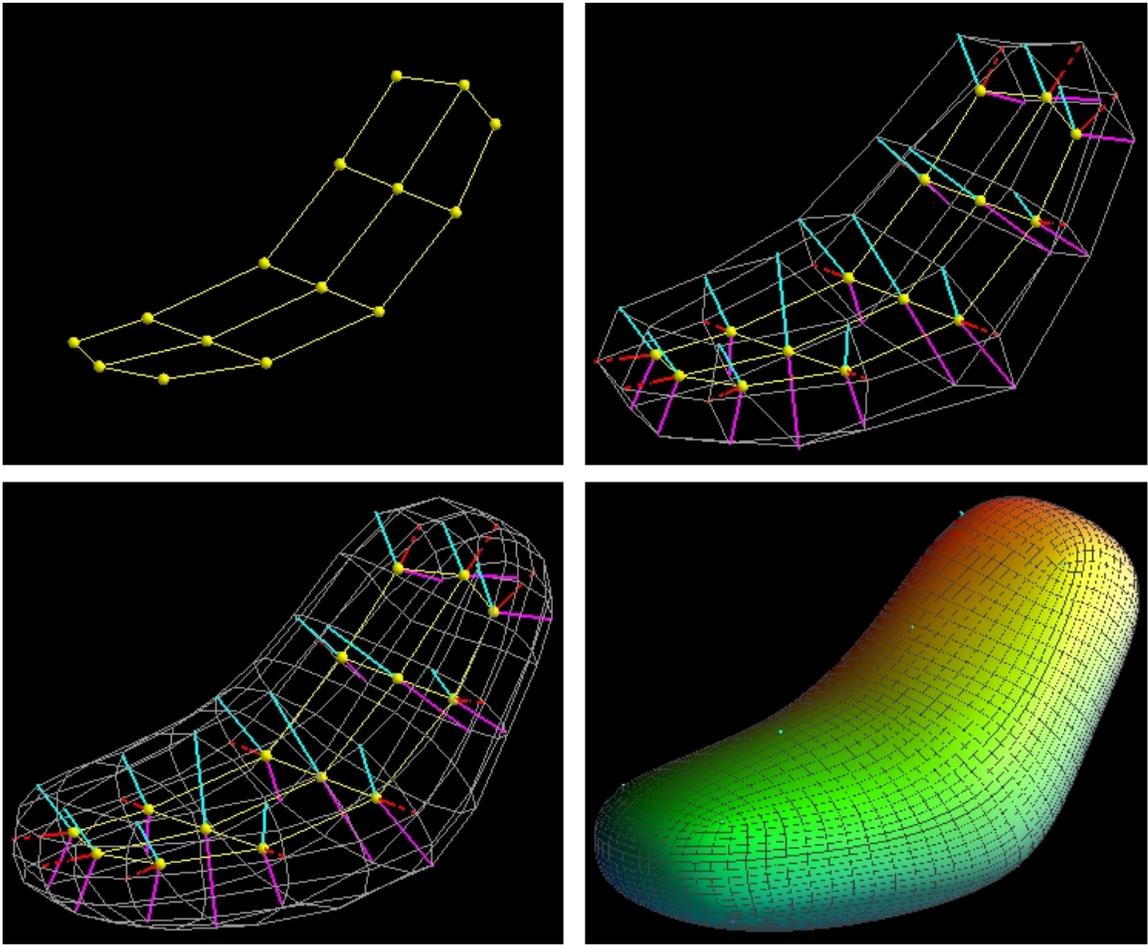


Figure 4.2: Medial mesh (upper left), boundary involutes and generating mesh (upper right), interpolating subdivision boundaries after one and three subdivisions (lower left and right, respectively).

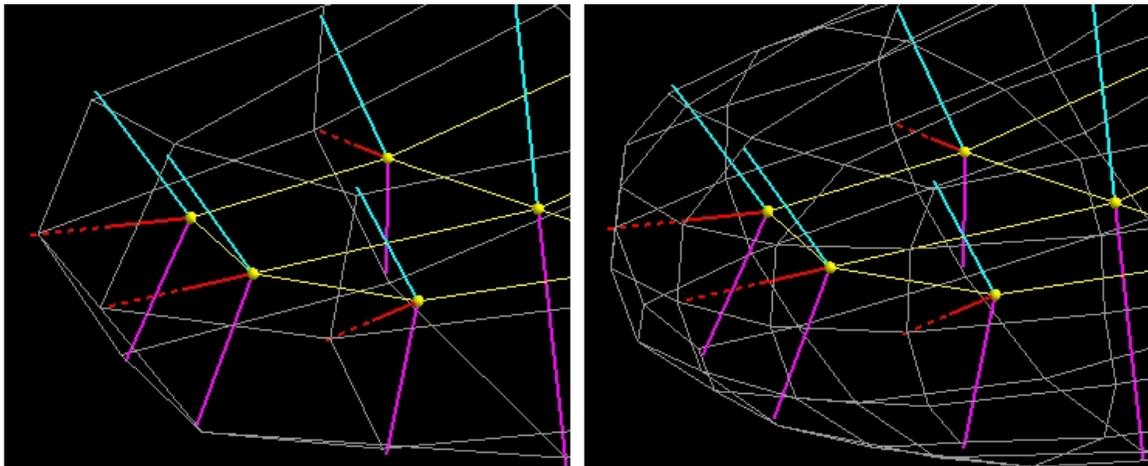


Figure 4.3: *Near*-interpolation of medial involutes by subdivision boundary.

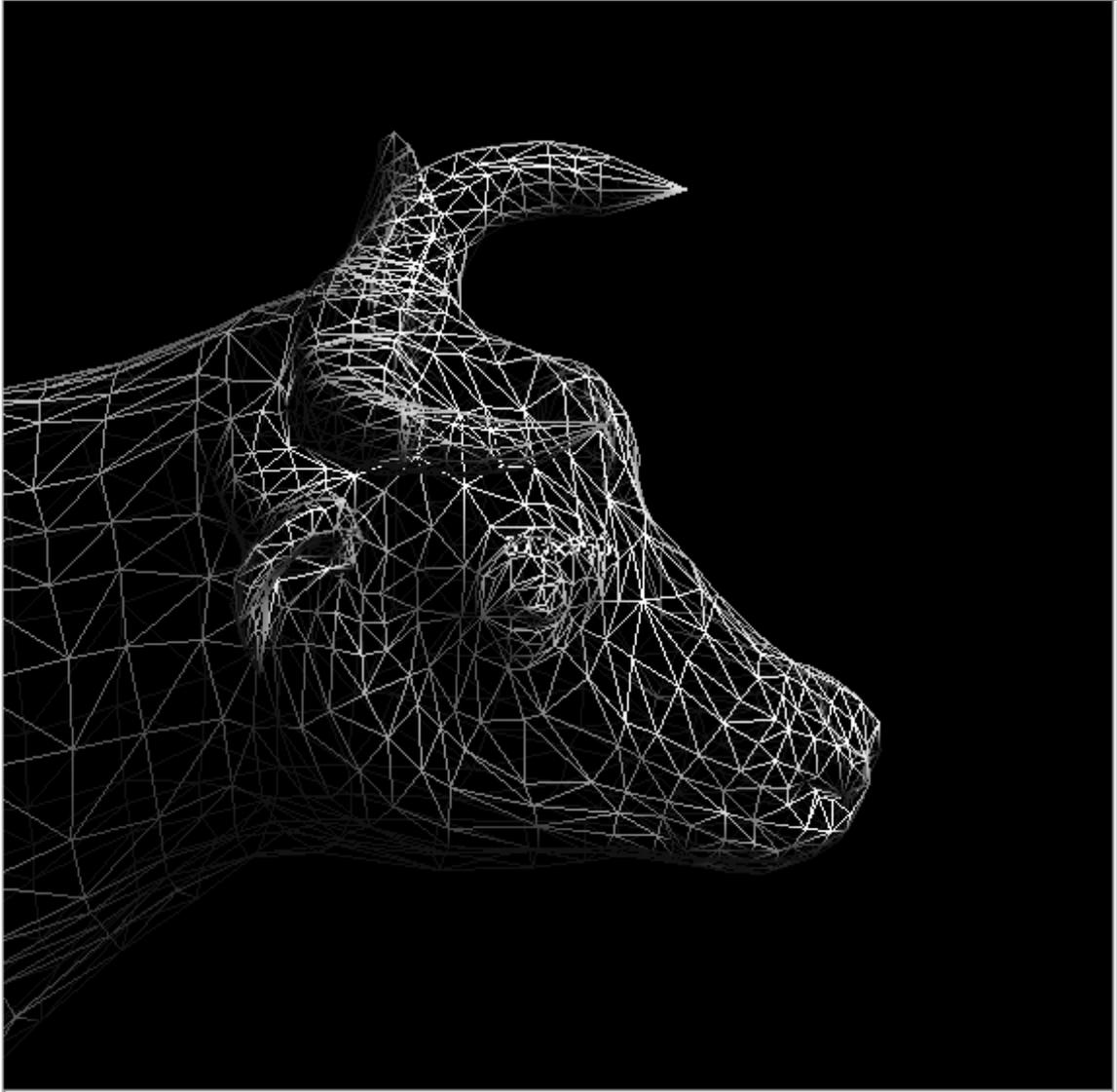


Figure 4.4: (a) Detail of an initial polygonal model `cow.ply`. This model was obtained from the Computer Graphics Group at the University of Virginia.

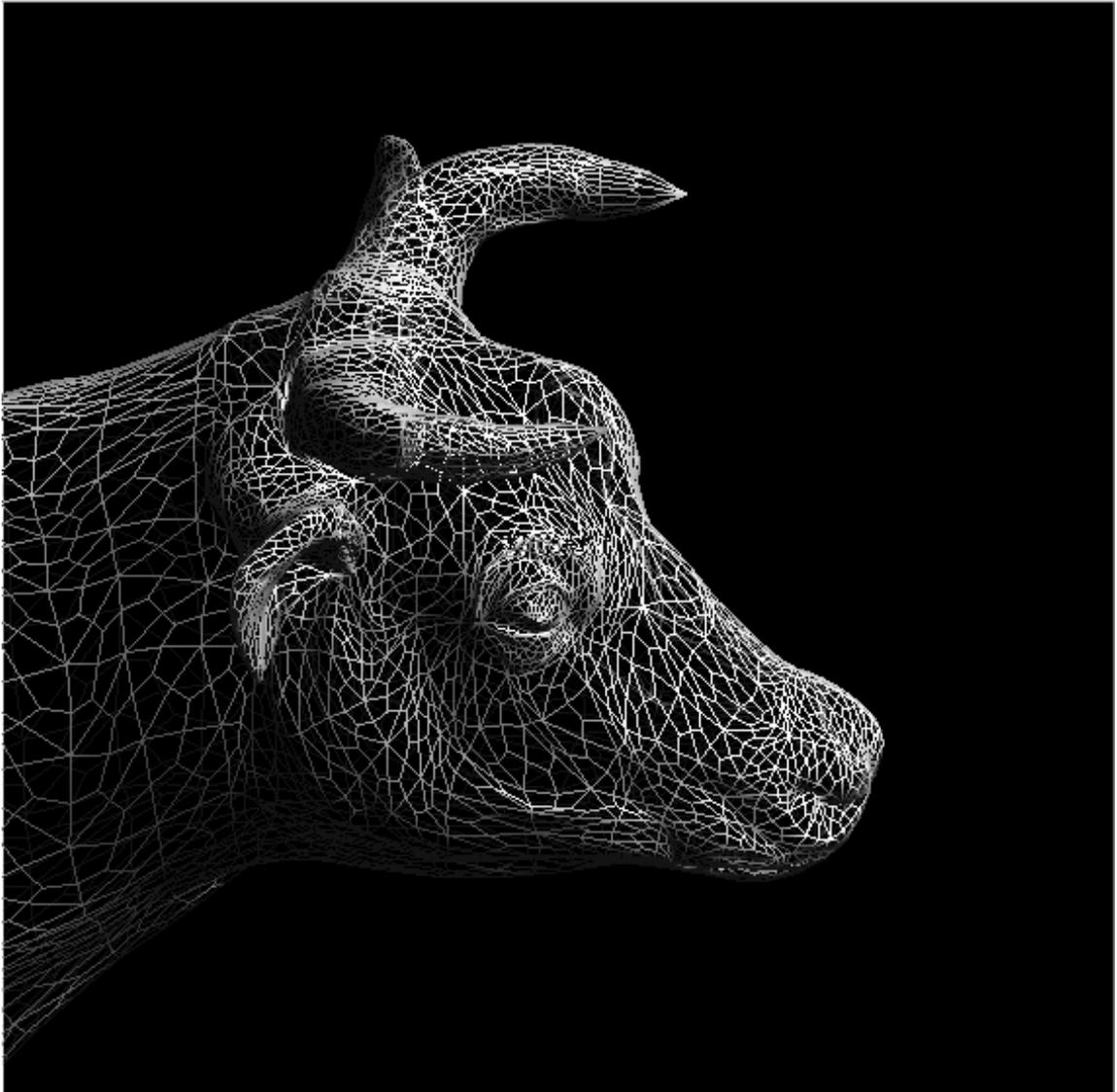


Figure 4.5: (b) One level of interpolating subdivision applied to the mesh of Fig. 4.4.



Figure 4.6: (c) Two levels of interpolating subdivision applied to the mesh of Fig. 4.4.

### 4.1.2 IIS-surfaces—regular vertices

Consider first an initial *regular* vertex mesh—one with only quadrangular faces. The IIS-surface is produced by creating a Catmull-Clark surface based on a modified initializing vertex grid; this modified grid is created by the following algorithm:

1. Initialize a boundary mesh  $v_n^0$  with vertices  $v_n^{\text{limit}}$  of positions to be interpolated
2. For iteration  $i = 1$  to  $m$
3. For each vertex  $v_j^i$  and its immediate edge  $e_{jk}^i$  and face  $f_{jk}^i$  neighbors
4. Compute perturbation  $\delta_j \ni: v_j^i + \delta_j$  gives  $v_j^{\text{limit}}$  as its limit position,  
based on current  $e_{jk}^i$  and  $f_{jk}^i$
5. Let  $v_j^{i+1} = v_j^i + \frac{1}{2}\delta_j$

Fig. 4.7 illustrates the indexing of vertices by  $k$  in a regular neighborhood of  $v_j^i$ . Step (4) is computed directly by solving the formula for the limit point given a vertex and its neighbors in an intermediate-level mesh. For a regular vertex of valence  $n$  and its  $2n$ -neighborhood

$$[v, f_1, f_2, \dots, f_n, e_1, e_2, \dots, e_n]$$

the limit point of Catmull-Clark subdivision is computed as

$$v_{\text{limit}} = \frac{1}{n(n+5)} \left[ n^2v + \sum_{k=1}^n [4e_k + f_k] \right]. \quad (4.1)$$

(Halstead [77]) Solving this for a perturbed  $v + \delta$  to produce a desired  $v_{\text{limit}}$  gives

$$v + \delta = \frac{n+5}{n} \left[ v_{\text{limit}} - \frac{1}{n(n+5)} \sum_{k=1}^n [4e_k + f_k] \right]. \quad (4.2)$$

Thus, given a mesh of limit positions, a perturbed mesh could be produced by substituting the perturbed  $v^i$  values. One would expect that such a perturbed mesh would over-correct for the expected vertex shifts under Catmull-Clark subdivision, since it ignores changes to a vertex's neighbors which would affect the actual subdivision. Therein lies the reasoning behind step (5), where the perturbation is averaged between  $v_j^i$  and the perturbed  $v_j^i + \delta_j$ . The above equation thus yields the iteration

$$\begin{aligned} v_j^{i+1} &= v_j^i + \frac{1}{2}\delta_j^i \\ &= v_j^i + \frac{1}{2} \left[ \frac{n+5}{n} \left[ v_{\text{limit}} - \frac{1}{n(n+5)} \sum_{k=1}^n [4e_k + f_k] \right] - v_j^i \right] \end{aligned} \quad (4.3)$$

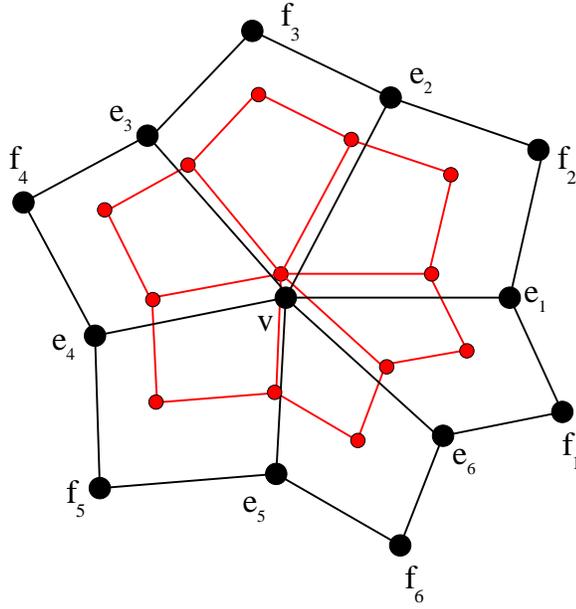


Figure 4.7: Vertex labeling in a regular 1-neighborhood. The  $e_i$  share an edge with the central vertex  $v$  and the  $f_i$  share a face.

for a vertex  $v_j$  of valence  $n$ . While this iteration appears messy, it actually amounts to a simple computation with a precomputed mask in the local 1-neighborhood of each vertex, thus requiring only local connectivity information on the initial involute mesh. Only one or two iterations produce a good approximation to the desired boundary; the iterative process, in fact, is computing a solution to the linear system defining the global problem by performing a successive overrelaxation (*SOR*) on a Jacobi iteration. To see that this is so, for the global subdivision, express the problem of solving

$$\begin{aligned}\vec{v}_{\text{limit}} &= \mathbf{A}\vec{v} \\ &= (\mathbf{D} + \mathbf{U} + \mathbf{L})\vec{v}\end{aligned}$$

and thus, elementwise,

$$\begin{aligned}v_{\text{limit}_j} &= A^j \vec{v} \\ &= (D^j + U^j + L^j) \vec{v} \\ &= D^j \vec{v} + (U^j + L^j) \vec{v} \\ &= \frac{n_j}{n_j + 5} v_j + \frac{1}{n_j + 5} \sum_{k=1}^n [4e_{kj} + f_{kj}].\end{aligned}\tag{4.4}$$

for  $\vec{v}$ , where  $\mathbf{A}$  is the (sparse) subdivision limit-surface matrix for the entire mesh and  $\vec{v}_{\text{limit}}$  is the vector of mesh vertices to be interpolated. As defined, each row  $j$  of the matrix  $\mathbf{A}$  computes the weighted sums in equation 4.1.2 for the  $v_j$  limit element of  $\vec{v}_{\text{limit}}$ . From this  $\mathbf{A} = (\mathbf{U} + \mathbf{L} + \mathbf{D})$  decomposes  $\mathbf{A}$  into its upper and lower triangular and its diagonal components, where the  $D^j$  component is the vertex weight for the  $v_j$  vertex, and the  $[U^j + V^j]\vec{v}$  gives the weighted sum of the edge and face neighbors of  $v_j$ . Inverting this to solve for  $v_j + \delta_j$  gives

$$\begin{aligned} v_j + \delta_j &= \frac{n_j + 5}{n_j} \left[ v_{\text{limit}_j} - \frac{1}{n_j + 5} \sum_{k=1}^n [4e_{kj} + f_{kj}] \right] \\ &= \frac{1}{D^j} [v_{\text{limit}_j} - (U^j + L^j)\vec{v}]. \end{aligned}$$

Thus, the solution for the linear system can be expressed as

$$\vec{v} + \vec{\delta} = \mathbf{D}^{-1} [\vec{v}_{\text{limit}} - (\mathbf{U} + \mathbf{L})\vec{v}] \quad (4.5)$$

where  $\vec{\delta}$  is the vector of perturbations  $\delta_j$  for each respective  $v_j$ .

Each iteration of equation 4.3 above can therefore be combined in the linear equation

$$\begin{aligned} \vec{v}^{i+1} &= \vec{v}^i + \frac{1}{2}\vec{\delta}^i \\ &= \vec{v}^i + \frac{1}{2} [\mathbf{D}^{-1} [\vec{v}_{\text{limit}} - (\mathbf{U} + \mathbf{L})\vec{v}^i] - \vec{v}^i] \\ &= \vec{v}^i + \frac{1}{2}\mathbf{D}^{-1} [\vec{v}_{\text{limit}} - (\mathbf{U} + \mathbf{L})\vec{v}^i - \mathbf{D}\vec{v}^i] \\ &= \vec{v}^i - \frac{1}{2}\mathbf{D}^{-1} [(\mathbf{U} + \mathbf{L} + \mathbf{D})\vec{v}^i - \vec{v}_{\text{limit}}] \\ &= \vec{v}^i - \frac{1}{2}\mathbf{D}^{-1} [\mathbf{A}\vec{v}^i - \vec{v}_{\text{limit}}] \\ &= \vec{v}^i - \frac{1}{2}\mathbf{D}^{-1}\vec{\xi}^i \\ &= \vec{v}^i - \omega\mathbf{D}^{-1}\vec{\xi}^i \end{aligned}$$

where  $\vec{\xi}^i$  is the error residual vector for  $\vec{v}^i$ . This is the canonical form for an SOR on a Jacobi iteration, as discussed in Press et. al. [125], Strang [143], and Golub and Van Loan [71]. (Actually, it is an underrelaxation, since  $\omega$  in the  $\omega\delta_j$  term is less than 1.) For diagonally dominant matrices, one expects good convergence for this method, and while  $\omega = \frac{1}{2}$  is a simple guess, the iteration converges so rapidly in test cases that

no fine-tuning was deemed necessary. The global subdivision matrix  $\mathbf{A}$  is not in fact diagonally dominant, and I have no formal proof of convergence. We know that the spectral radius is 1, and the 2nd and 3rd eigenvalues are 0.5. The algorithm has not failed to converge for any closed, two-sided m-rep boundaries to which it has been applied.<sup>2</sup>

To further illustrate this method, consider the cases of regular vertices of valence 3 and 4. For an ordinary even vertex and its 8-neighborhood

$$[v, f_1, f_2, f_3, f_4, e_1, e_2, e_3, e_4]$$

the limit point of Catmull-Clark subdivision can be computed as

$$v_{\text{limit}} = \frac{4}{9}v + \frac{1}{9} \sum_k e_k + \frac{1}{36} \sum_k f_k \quad (4.6)$$

and solving for a perturbed  $v + \delta$  producing a given  $v_{\text{limit}}$  gives

$$v + \delta = \frac{9}{4} \left[ v_{\text{limit}} - \frac{1}{9} \sum_k e_k - \frac{1}{36} \sum_k f_k \right]. \quad (4.7)$$

Similarly, for a valence-3 vertex with its 6-neighborhood, one has

$$v_{\text{limit}} = \frac{3}{8}v + \frac{1}{6} \sum_k e_k + \frac{1}{24} \sum_k f_k \quad (4.8)$$

and thus

$$v + \delta = \frac{8}{3} \left[ v_{\text{limit}} - \frac{4}{9} \sum_k e_k - \frac{1}{9} \sum_k f_k \right]. \quad (4.9)$$

The formulas in equations 4.7 and 4.9 yield the iterations

$$\begin{aligned} v_j^{i+1} &= v_j^i + \frac{1}{2} \delta_j^i \\ &= v_j^i - \frac{1}{2} \left[ v_j^i + \frac{9}{4} \left[ \frac{1}{9} \sum_k e_k^i + \frac{1}{36} \sum_k f_k^i - v_{j\text{limit}} \right] \right] \end{aligned} \quad (4.10)$$

---

<sup>2</sup>Given that a Jacobi iteration converges on an acceptable limit boundary, one might wonder if a Gauss-Seidel approach would be equally effective. Preliminary experiments show that this is the case. Gauss-Seidel has advantages over Jacobi, as well, in allowing vertex perturbations to be made in-place, thus giving both faster convergence and using less memory.

and

$$\begin{aligned}
v_j^{i+1} &= v_j^i + \frac{1}{2}\delta_j^i \\
&= v_j^i - \frac{1}{2} \left[ v_j^i + \frac{8}{3} \left[ \frac{1}{6} \sum_k e_k^i + \frac{1}{24} \sum_k f_k^i - v_{j\text{limit}} \right] \right]
\end{aligned} \tag{4.11}$$

for a vertex  $v_j$  of valence 3 or 4 respectively.

Special issues are raised by the presence of non-quad faces in the initializing mesh. Given a vertex with adjacent faces with greater or fewer than 4 sides, one would like to avoid special cases in the formulae for inverting the limit point equations, which themselves would have to be based on an altered eigenstructure for the subdivision. Traditionally, limit positions at irregular vertices are computed by first performing a single Catmull-Clark splitting and averaging, which results in a regular mesh, and then applying the limit masks to the new vertices. Attempts to invert this two-stage approach are complicated by the arbitrary number of vertices possible in neighboring polygons of a given mesh vertex. Instead, as will be shown below, it is possible to *regularize* the neighborhood about a vertex to create a regular 1-neighborhood with the same limit-structure. Given such a regularized neighborhood, the standard limit-position and limit-tangent masks can be applied, and the above inversions and iterative interpolation method can be applied without modification. This technique will be discussed below.

### 4.1.3 IIS-surfaces—irregular vertex limit positions and inverses

Recall that an *irregular* mesh is one with non-quadrangular polygons, and an irregular vertex is one at the corner of such a polygon. Because an initial Catmull-Clark mesh (such as one produced by blending a figure and subfigure) may contain irregular vertices, it is necessary to compute their limit points as well. This is frequently done in two stages:

1. a single Catmull-Clark split-and-average, after which all vertices are regular, followed by
2. the application of the limit mask for a regular vertex at the level-2 vertices.

This is inadequate for the needs of interpolating subdivision as developed above, where the limit equations must be inverted for use in the iterative, SOR interpolation

scheme. Instead, going back to first principles for Catmull-Clark subdivision and working again from Halstead et al. [77], one can directly compute the limit point and tangent vectors for an irregular vertex, either ordinary or extraordinary. There is no standard matrix form for the subdivision at a vertex with an irregular one neighborhood. Instead, Catmull-Clark subdivision proceeds equivalently in the following way.

1. New face vertices  $f_1^{i+1}, \dots, f_n^{i+1}$  are created at the centroid (arithmetic mean) of the bounding polygon vertices for each bounding polygonal face  $P_1, \dots, P_n$ .
2. New edge vertices are created

$$e_j^{i+1} = \frac{v^i + e_j^i + f_{j-1}^{i+1} + f_j^{i+1}}{4}. \quad (4.12)$$

3. The new vertex  $v^{i+1}$  can now be computed as

$$v^{i+1} = \frac{n-2}{n}v^i + \frac{1}{n^2} \sum_j e_j^i + \frac{1}{n^2} \sum_j f_j^{i+1} \quad (4.13)$$

After the initial splitting, the new mesh is and remains regular. In regard to vertex valences, extraordinary vertices will be created as new face-vertices of non-quadrilateral polygons, but all subsequent splittings leave vertex valences unchanged and create no new extraordinary vertices.

In the above algorithm, for a vertex  $v^0$  having an irregular 1-neighborhood, the new 1-neighborhood  $\{v^1, f_1^1, f_2^1, \dots, e_1^1, e_2^1, \dots\}$  depends only on  $\{v^0, f_1^1, f_2^1, \dots, e_1^0, e_2^0, \dots\}$ . That is, the only difference an irregular mesh element makes is in the computation of the  $f_i^1$  face vertices for the respective  $P_i$  polygons. Let  $|P_i|$  be the number of vertices in polygon  $P_i$ .

**Theorem:** Assume that  $P_i$  is in a 1-neighborhood of  $v^0$  that contains edges to vertices  $e_i^0$  and  $e_{i+1}^0$ . If  $|P_i| \neq 4$ , a new polygon  $P_i'$  can be constructed such that

1.  $|P_i'| = 4$ ,
2.  $P_i'$  contains  $v^0$  and the edge-vertices  $e_i^0$  and  $e_{i+1}^0$ , and
3.  $P_i'$  has the same centroid as  $P_i$ .

The proof is fairly trivial.<sup>3</sup> If  $|P_i| = n > 4$ , the vertices of  $P_i$  can be ordered as  $\{v^0, e_i, c_1, c_2, \dots, c_{n-3}, e_{i+1}\}$ . Then it is necessary to find an  $P'_i = \{v^0, e_i, R, e_{i+1}\}$  having the same  $f_i^1$  as its centroid, i.e.,

$$\begin{aligned}
 f_i^1 &= \frac{1}{n} \left[ v^0 + e_i + e_{i+1} + \sum_{j=1}^{n-3} c_j \right] \\
 &= \frac{1}{4} [v^0 + e_i + e_{i+1} + R] \\
 \Rightarrow \\
 R &= \frac{4}{n} \left[ \sum_{j=1}^{n-3} c_j \right] - \frac{n-4}{n} [v^0 + e_i + e_{i+1}]. \tag{4.14}
 \end{aligned}$$

For the case  $|P_i| = 3$ ,

$$\begin{aligned}
 f_i^1 &= \frac{1}{3} [v^0 + e_i + e_{i+1}] \\
 &= \frac{1}{4} [v^0 + e_i + e_{i+1} + R] \\
 \Rightarrow \\
 R &= \frac{1}{3} [v^0 + e_i + e_{i+1}],
 \end{aligned}$$

and the new  $R$  is simply the centroid of the three vertices in  $P_i$ . This is simply Eqn. 4.14 for case  $n = 3$ , with  $\sum c_j$  nil, and not really a special case. Fig. 4.8 gives an illustration of the process.

**Corollary:** An irregular 1-neighborhood of a vertex  $v^0$  can be replaced by a regular neighborhood producing the same  $\{v^1, f_1^1, f_2^1, \dots, e_1^1, e_2^1, \dots\}$  and therefore having the same limit structure (position and tangent space) in the neighborhood of  $v^{\text{limit}}$ .

This corollary follows directly by applying the above method to all irregular polygons adjoining a vertex, given the known facts:

1. the level- $(n + 1)$  1-neighborhood for an ordinary or extraordinary vertex  $v^{n+1}$  is determined completely by  $v^n$ , by the level- $n$  edge vertices  $e_i^n$ , and by the level- $(n + 1)$  face vertices  $f_i^{n+1}$ ;
2. the limit structure can be derived explicitly by eigenanalysis, given a regular 1-neighborhood about an ordinary or extraordinary vertex.

---

<sup>3</sup>Feynman's Observation: anything that can be proven is trivial.

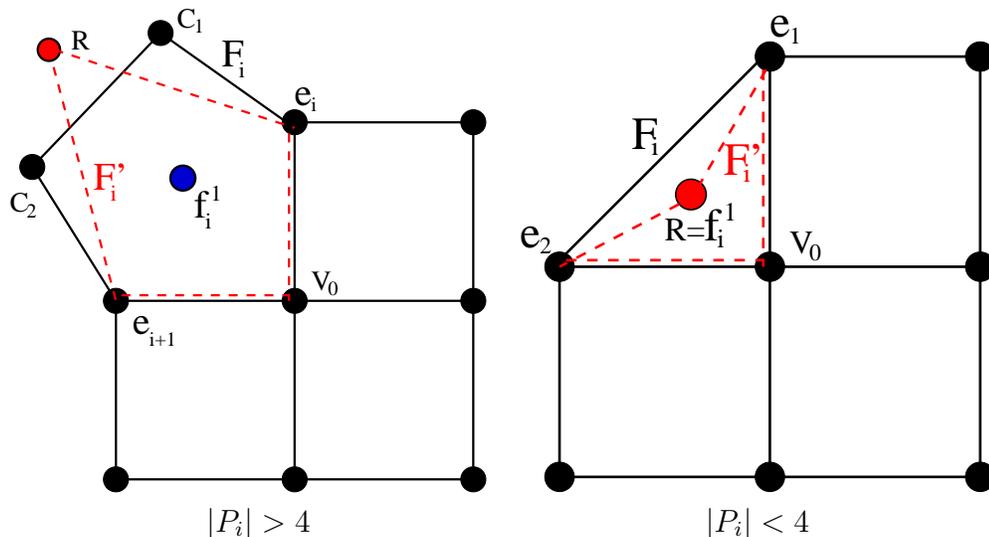


Figure 4.8: Regularizing the neighborhood of an irregular vertex.

These results show that a per-vertex mesh regularization allows limit positions to be determined directly from the initial mesh positions; this, in turn, allows the iterative SOR technique be used to interpolate original mesh positions for irregular vertices just as for regular ones.

Figs. 4.9 and 4.10 illustrate the iterative interpolation applied to two different initial meshes; the former shows the algorithm applied to a coarse genus-1 mesh; the latter illustrates the algorithm's effectiveness given triangular faces in a mesh. (Figs. 4.5 and 4.5 show the algorithm applied to a triangulated mesh, as well.) These images illustrate the rapid convergence of the iteration on the initializing mesh positions.

#### 4.1.4 Error metrics for IIS-surfaces relative to medially implied boundaries

IIS-surfaces do not interpolate mesh normals; thus, the  $\vec{v}_1$ ,  $\vec{v}_2$ , and  $\vec{b}$  vectors of the medial atoms defining the mesh of boundary involutes will not match the surface normals of the interpolating subdivision meshes. The degree of mismatch can, in fact, be arbitrarily bad. In most cases, the mesh topology for the closed surface at the coarsest scale produces surface normals acceptably near to the values implied by the medial atoms. Fully interpolating surface subdivision, as typified by Halstead, require extra conditions to prevent rippling effects on interpolating surfaces. Because the interpolation mesh for an m-rep is computed only for the coarsely sampled, medially

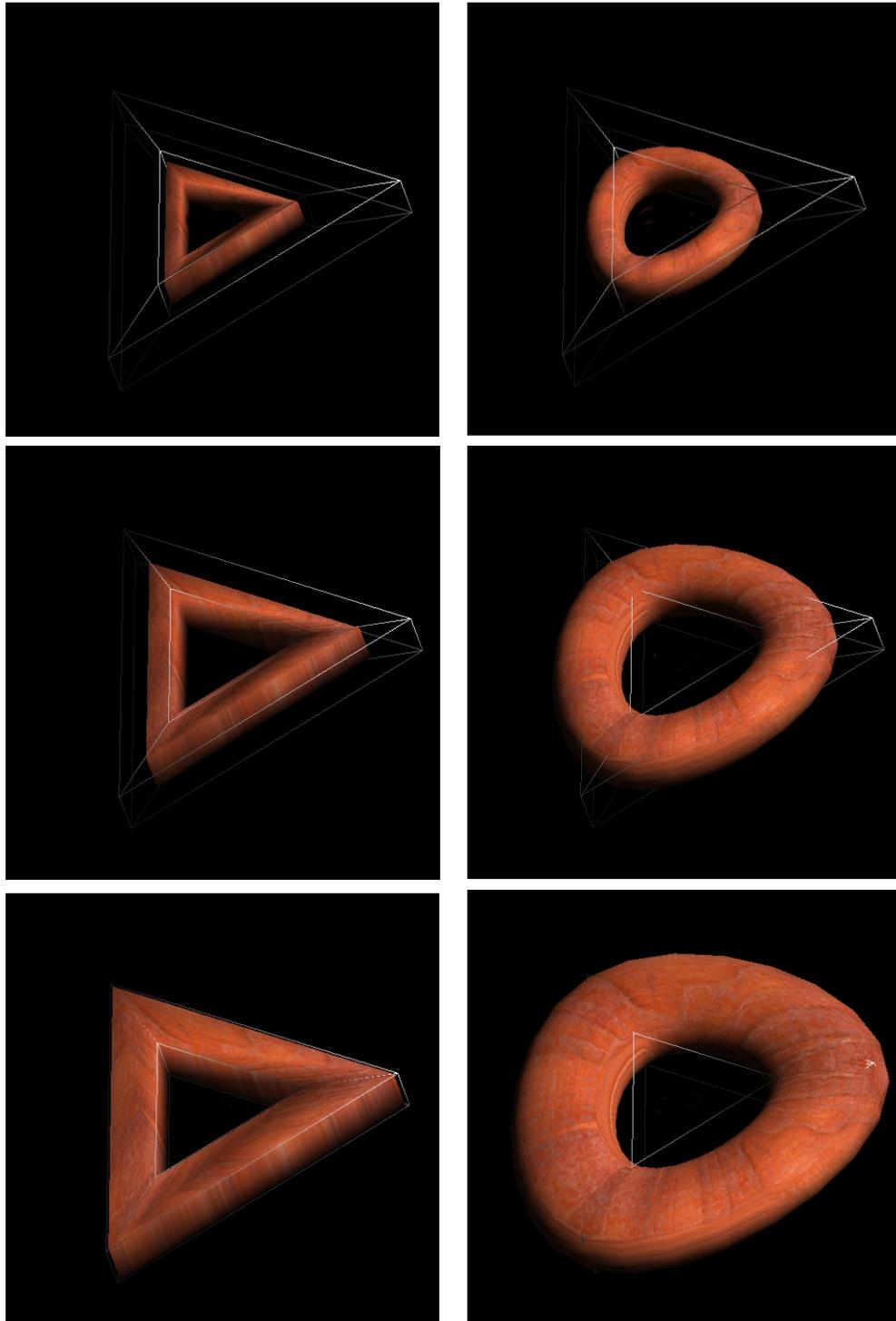


Figure 4.9: Iterative subdivision applied to a coarse torus-mesh. The images on the left show the limit points of the initial mesh after zero, one, and three iterations of the interpolation. The images on the right show corresponding subdivision surfaces for the respective iterations. The outlining shows the initial mesh positions.

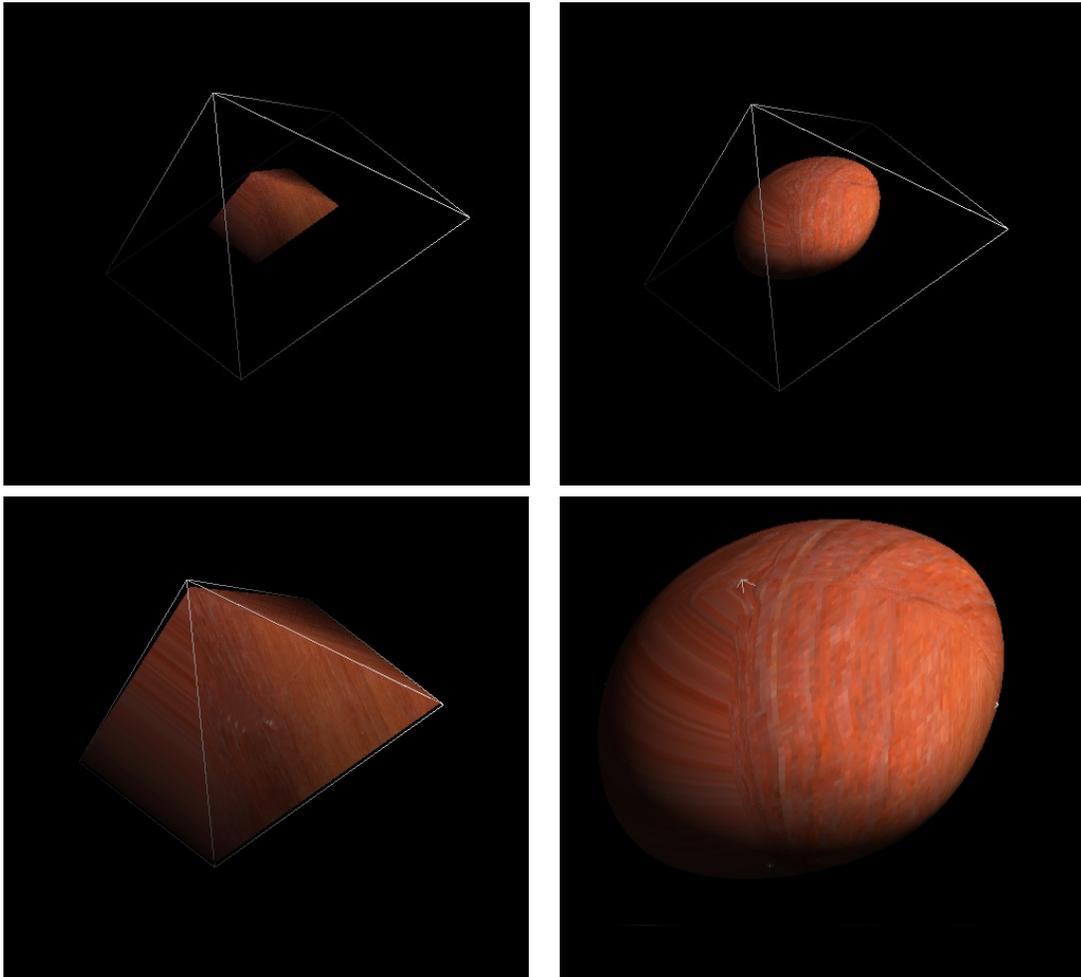


Figure 4.10: Iterative subdivision applied to a pyramid-mesh. The images on the left show the limit points of the initial mesh after zero and three iterations of the interpolation. The images on the right show corresponding subdivision surfaces for the respective iterations. The outlining shows the initial mesh positions.

implied boundary, and because the normals are not specified explicitly, the smoothing inherent in stationary subdivision is sufficient to avoid such rippling in typical cases. A method of as-needed normal interpolation has been developed; this method is fast and simple, but it lacks the fine control of more sophisticated methods that include bending-energy minimization, and it tends to exacerbate the problem of ripples, which must then be eliminated using small-scale boundary-texturing displacements.

To produce an as-needed normal-adjusting method, a metric is first required to quantify the deviation of surface normals from medially implied normals; a non-stationary modification to the subdivision algorithm can then be applied when deviation exceeds some threshold. To compare initial interpolating mesh locations and normals with their corresponding locations and normals given by the medial mesh atoms, there are several relevant metrics:

1. measurement of the distance from an original involute endpoint to its corresponding position in the interpolating mesh, scaled by  $r$ ;
2. measurement of the rotation  $\theta$  from an original involute normal to its corresponding subdivision boundary normal;
3. measurement of the ( $r$ -proportional) shift from a known medial location  $\mathbf{p}$  to a new  $\mathbf{p}'$  based on creating a new medial atom from associated boundary locations and normals.

Metric (1) has been used to determine how many iterations are needed to approximate the medially implied boundary, by providing an  $r$ -proportional tolerance on the accuracy of the iterative interpolation. Metric (2) has been used to determine if and where a non-stationary normal-interpolation method should be applied to the subdivision mesh. The Lie group transformations of Fletcher [56] show that changes to  $\theta$  at the boundary should also be measured in units of  $r\theta$ ; thus, boundary rotations of equal  $\theta$  are more significant in wider regions. Metric (3) has been used when interpolating medial atoms and is based on the involute-to-average-medial-atom technique presented in the last chapter (recall Fig. 3.17). The  $r$ -proportional distance of displacement of the old medial location  $\mathbf{p}$  from the new  $\mathbf{p}'$  gives a measure of how poorly the coarse medial sampling fits the interpolating surface in a Blum-sense.

### 4.1.5 Accurate normal interpolation by modified IIS-surfaces

In many cases, the basic topology and structure of the coarse involute mesh guides the subdivision to approximately correct normal values. Nonetheless, without a boundary fit that incorporates the surface normals implied by the medial atoms, the m-rep medial atom to boundary-implied medial atom error can be made arbitrarily bad. Using Metric (2) above, it is simple to identify if and where normal deviation falls outside of acceptable bounds. For such situation, I have created a simple method to interpolate the normal at a given location, using a non-stationary, non-uniform modification to the interpolating subdivision algorithm. **Given:** an initial mesh of medially implied boundary involutes  $v_n$  with implied normals  $\vec{n}_n$ ,

1. Construct a level-0 IIS-surface mesh of vertices  $v_n^0$  that give  $v_n^{\text{limit}} \approx v_n$  under Catmull-Clark subdivision.
2. Subdivide twice to get vertices  $v_n^2$  and their level-2 1-neighborhoods.
3. Compute  $v_n^{\text{limit}}$  and  $\vec{n}_n^{\text{limit}}$ .
4. For  $i = 1$  to  $n$  do
  5. If  $\vec{n}_i^{\text{limit}}$  is acceptably close to  $\vec{n}_i$ , **Done**.
  6. Else, compute the rotation  $\mathcal{R}_i \in SO(3)$  taking the vector  $\vec{n}_i^{\text{limit}}$  to  $\vec{n}_i$ .
  7. Rotate  $v_i^2$  and its entire 1-neighborhood by  $\mathcal{R}_i$  about  $v_i^{\text{limit}}$ .
  8. Substitute these rotated vertices into the  $v_n^2$  mesh.

The resulting modified  $v_n^2$  mesh will interpolate the normals for the limit positions of the modified vertices. The proof of this is trivial, resting on three facts:

- local 1-neighborhoods of initial vertices are disjoint after 2 subdivisions;
- a vertex's limit-surface position and normal depend only on the 1-neighborhood of that vertex;
- subdivision is rotationally invariant, allowing the rotation of the limit normal to be achieved by rotating the subdivision neighborhood by the same amount about the limit point while keeping that limit point unchanged.

It is a strength of this method that it is applied on an *as needed* basis and that it requires only an  $\mathcal{O}(1)$  operation for those of the  $n$  initial vertices requiring modification, adding at worst  $\mathcal{O}(n)$  operations overall (for  $n$  typically small). Further, the interpolation can be applied in-place on a subdivision mesh, modifying only the independent 1-neighborhoods about the selected mesh positions.

Fig. 4.11 shows an m-rep boundary generated by this algorithm; the polygonal outlines show where the underlying subdivision mesh was perturbed to match normals in the neighborhoods of the sampled medial involutes. An advantage of the normal interpolating method is that the sampled medial atoms now form an accurate Blum-medial location for the generated boundary and thus provide the correct  $\nabla r$  information for a sampled medial representation. Using a non-normal-interpolating IIS-surface essentially treats the medial skeleton as a sampled chordal axis, since the 1st-order information provided by the medial atoms is ignored. For an m-rep with sparse surface sampling, the normal interpolation method generally gives good results. More problematic are m-rep models based on previous non-normal-interpolating boundaries, especially cases in which medial atoms were positioned by hand or by optimization methods. Normal-interpolation in such cases can lead to severe rippling effects; Fig. 4.12 illustrates this, comparing a bone (in a pelvic-region model) with boundary generated by both normal-interpolating and non-normal-interpolating IIS-surfaces.

This method is thus subject to the same potential difficulties with surface ripples that afflict other subdivision techniques that interpolate normals. To reduce rippling, it might be useful to space the rotation out over several subdivision steps to give a smoother interpolation—this is similar to a technique developed by Biermann[9]. It is also possible that ripples can be reduced by doing a *minimal* perturbation so that the boundary normal is just within a specified tolerance without being exactly interpolating. In addition, one might do a smoothing on the mesh after perturbing  $v^2$  and its neighborhood. There remains a ring of unperturbed vertices around each perturbed neighborhood; these can be moved to reduce rippling effects. A simple way to accomplish this is to perform a standard Catmull-Clark subdivision of the boundary after the perturbation step; although this will be non-interpolating, the shrinkage of the model from the limit positions after two initial subdivisions is typically minor and not  $r$ -proportionately significant. For m-rep work, the boundary perturbation technique has been fairly effective and computationally cheap. As alternatives, however, there remain the general interpolating subdivision schemes such as Halstead’s or Biermann’s that give more precise control over boundary normals and curvature.

#### 4.1.6 Interpolation of other boundary attributes

The usefulness of a medially defined coordinate system was discussed in the previous chapter in Sec. 3.1.5. It is necessary, therefore, to interpolate medial coordinates from known involute positions on the boundaries and to interpolate approximate

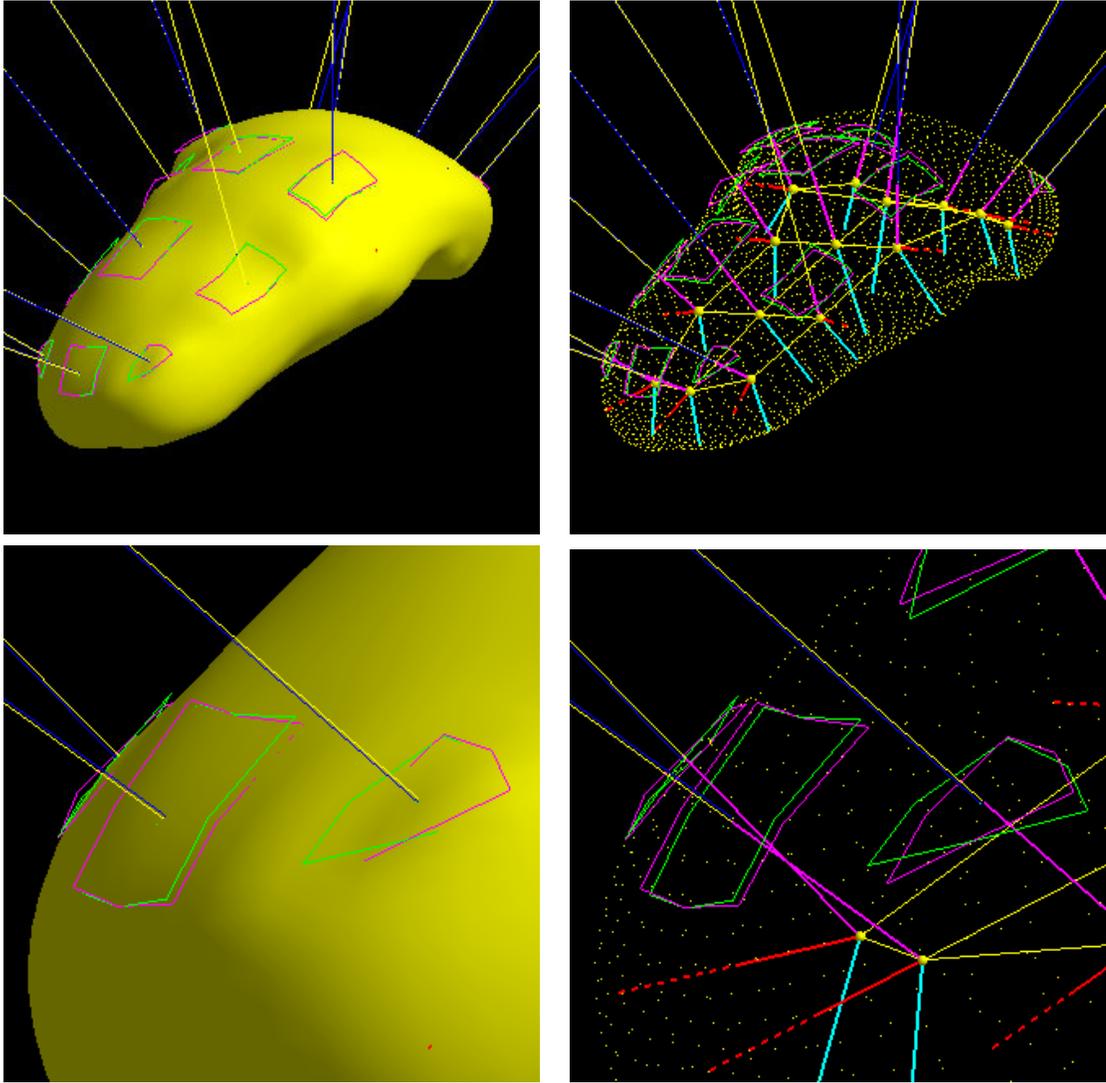


Figure 4.11: An m-rep boundary generated by normal-interpolating IIS-surfaces. The polygonal regions show the perturbation of 1-neighborhoods in the underlying subdivision mesh to match boundary normals at the sampled medial involutes.

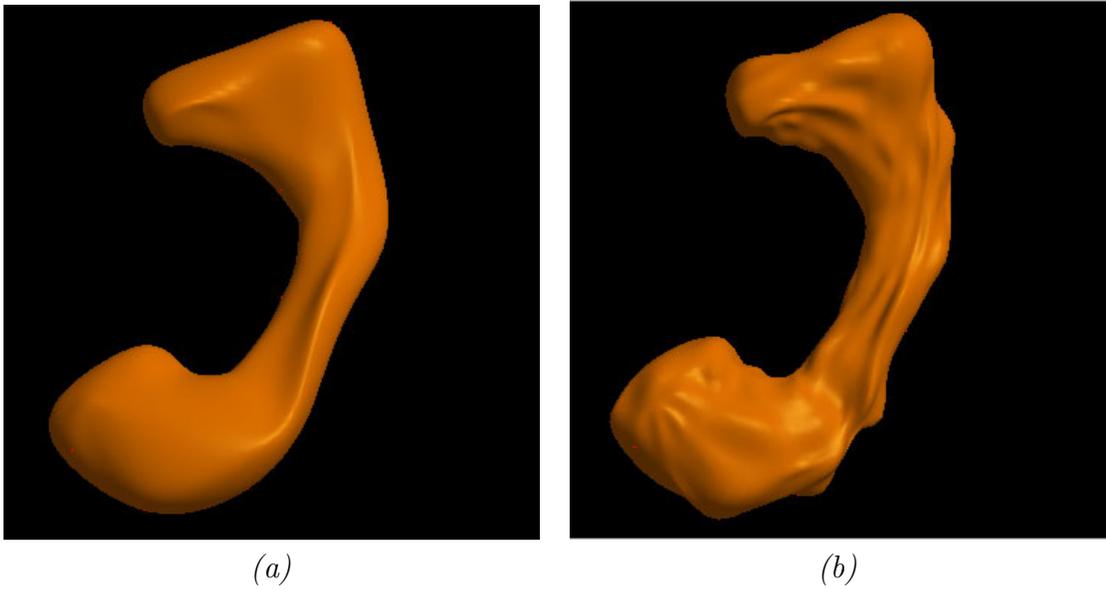


Figure 4.12: A bone modeled by (a) ordinary, and (b) normal-interpolating IIS-surfaces. Medial atoms in this model were placed without regard for the boundary normals; thus, the rippling produced by normal interpolation is severe.

values of  $r$  as well. The standard practice for texture coordinate interpolation on subdivision surfaces is to split the texture coordinates at the same time the mesh itself is split. Typically, texture coordinates will be subdivided using the same splitting and perturbation scheme as for the vertices (DeRose[43]); this gives  $2^{nd}$  order continuity to the surface coordinates. However, for an interpolating surface such as an m-rep, this results in “coordinate-creep”, with values shifting from their known values at interpolated positions.

Instead, a simple midpoint subdivision rule has been used for  $r$  and the  $(u, v, t)$  coordinates, leaving values unchanged at even vertices. The values at odd edge-vertices are the average of the even endpoints, and the values at odd face-vertices are the average of the even face vertices. This does have the drawback of giving only  $0^{th}$  order continuity of the coordinate fields on the surface; this has not proven to be a sticking point for any of the current research. If parametric continuity is needed, the IIS-surface interpolating scheme used on the surface might be used to interpolate the other surface attributes.

The use of a subdivision scheme to create an  $r$ -field on the surface provides an approximation to the true medial radius function on the surface; this field can be used to determine width-proportional tolerances at boundary locations. This matter will be discussed in the next chapter in Sec. 5.1. While subdivision-based coordinate

interpolation allows coordinates to be assigned to all mesh vertices at any level, it does not solve the problem of finding an  $(x, y, z)$  location in space corresponding to an arbitrary  $(u, v, t)$  surface position. Methods for this will be discussed in detail in Sec. 6.2; they basically involve linearly interpolating vertex parameter values across tiled boundaries at a subdivision level deemed suitably fine.

#### 4.1.7 Drawbacks and limitations of IIS-surfaces

There are a number of drawbacks in terms of using subdivision surfaces for m-rep boundaries. They lack explicit parameterizations and thus closed-form surface curvatures, principal directions, and fundamental forms, which would be useful for doing differential geometry on them. One can fit splines to regular regions or do Stam-style parameterization [137]; one might use methods of Jörg Peters and Georg Umlauf for finding Gaussian and mean curvature of subdivision surfaces [117, 118], or surface-mesh-based approaches as per Desbrun et al. at the Caltech Multi-Res Modeling Group [44]. It remains a fact that analysis on subdivision boundaries is non-trivial.

These are drawbacks for *all* subdivision surfaces. There is a drawback particular to iteratively interpolating subdivision surfaces. A standard technique for putting edges and creases in Catmull-Clark or other subdivision surfaces is to “freeze” certain vertices and edges in the mesh, allowing averaging only along a restricted set of mesh edges or not at all. This method cannot be applied directly to the modified meshes produced by IIS-surfaces; it might be possible to freeze vertices and edges in the initial mesh and not do inverse iteration there, but I suspect that this will produce undesirable artifacts. Boundary displacement, as discussed in the next chapter, makes vertex/edge freezing less necessary; whether it eliminates the need entirely remains to be seen.

Another drawback common to all surface-fitting by stationary subdivision is that the surface-fit is essentially a bicubic-spline interpolation, restricting the boundary curvature behavior accordingly. Thus, as Fig. 4.13 illustrates, a surface fit to an endcap (the mostly highly curved region, typically) may have undesirable wiggles. The use of the edge-atom  $\eta$ -elongation can resolve this, or a boundary perturbation might do likewise; the last recourse is a subdivision-surface-fitting algorithm with boundary curvature constraints, such as Halstead’s.

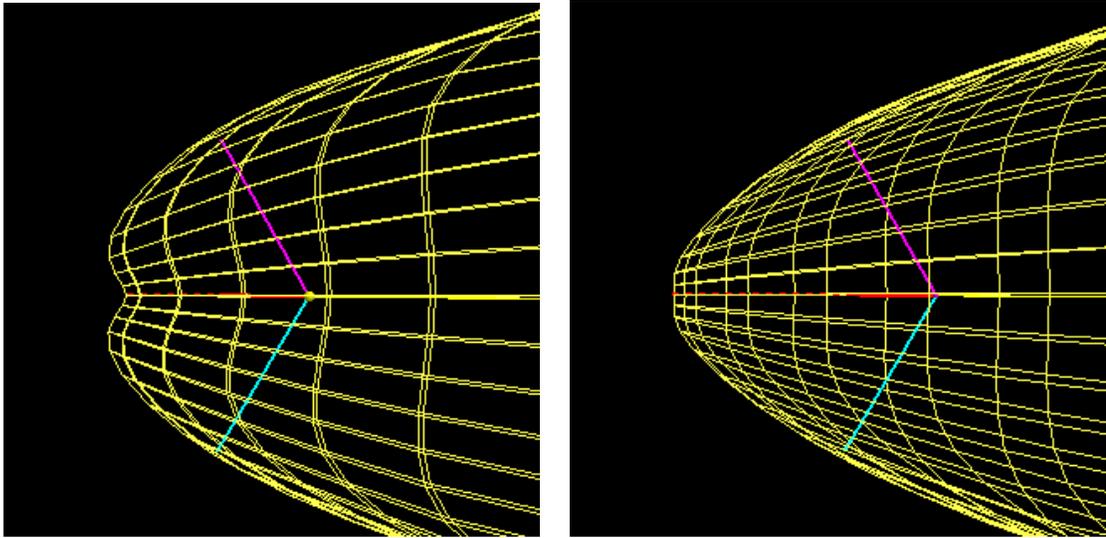


Figure 4.13: An endcap interpolated using interpolating IIS-surfaces. On the left, with  $\eta = 1$ , the surface has typical cubic-spline wiggling. On the right,  $\eta$  has been adjusted and the medial atom moved to eliminate the effect.

## 4.2 Proximity tests on m-rep boundaries

Given the medial-to-boundary correspondence of m-rep models, one would like the ability to compute the  $(u, v, t)$  boundary position of the nearest point on the surface to a test point in space. Such a method was needed to address several driving problems:

1. the definition of an implicit surface function for figure-subfigure blending, as previously implemented in Pablo for subfigure attachment (see Sec. 6.2);
2. the discovery of corresponding medial positions and opposing medial involutes for a given boundary position and its normal (see Sec. 4.3);
3. finding figure-subfigure intersection points for remeshing in blend regions (see Sec. 5.2).

The task of finding nearpoints on a subdivision surface to an arbitrary point in space is an open problem in multiresolution surface research. For m-rep research a method was developed that, while lacking the robustness of a general solution to the subdivision surface nearpoint problem, gave an acceptable approximation to the  $(u, v, t)$  coordinate of a nearpoint on an m-rep boundary. The intention was to limit the subdivision-level of the surface and still compute an approximate  $(u, v, t)$  correspondence for the limit-surface. One of the goals was to achieve a high enough query

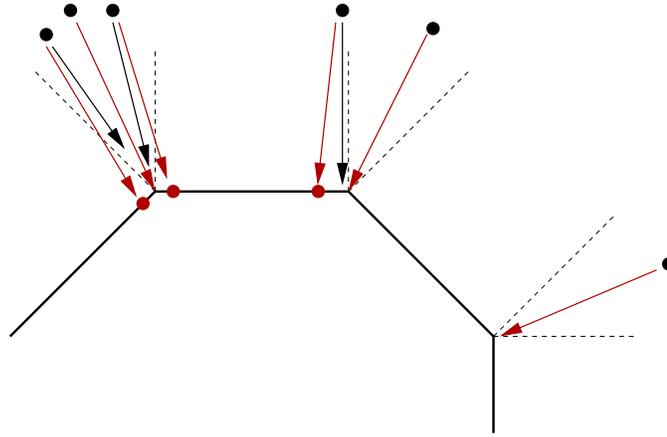


Figure 4.14: Euclidean minimum distance vs. *Phong-normal* projection nearpoint. The red arrows indicate Phong-normal nearpoints; the black arrows indicate Euclidean nearpoints.

rate to make the test suitable for interactive, real-time simulations. In this way, for instance, deformable m-rep models might be manipulated and modified using haptic interfaces without the need to subdivide at a deep level.

The method developed for IIS-surfaces is based on using a Phong-interpolated normal to approximate surface normals across subdivision mesh faces. These approximate normals are used in a minimization algorithm to find surface positions with interpolated normals that intersect the target point in space. The method can be applied to any subdivision surface which is either interpolating or for which limit points and limit normals can be computed for mesh vertices at each subdivision level.

#### 4.2.1 The Phong-normal nearpoint estimate

For a coarsely subdivided subdivision mesh, a Euclidean nearest-vertex/edge/face approach fails to take into account implied curvature information regarding the underlying surface. The approach explored here is to use the known surface normals at mesh vertices to approximate surface normal directions using bilinear interpolation—Phong-interpolation—of the surface normals. Fig. 4.14 illustrates in  $2D$  the difference between *Phong-normal* nearpoints and ordinary Euclidean ones. In the limit for a subdivision surface, as the boundary is subdivided to a finer and finer mesh, this will give a closer and closer approximation to a true local nearpoint, as the normals at the grid vertices converge on the surface normal for the infinitesimal surface region bounded by the vertices. The point found is a *local* nearpoint approximation: this

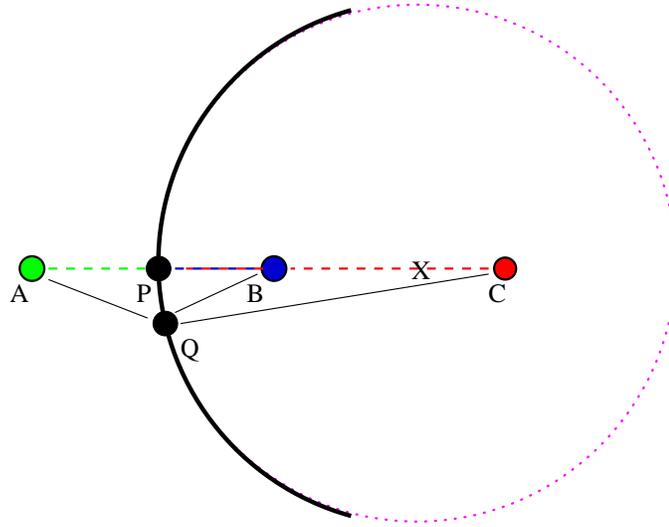


Figure 4.15: A nearpoint defined by normal projection may actually be a farpoint. In this illustration, **A**, **B**, and **C** all lie along the surface normal at point **P**. Point **C**, however, lies beyond the radius of curvature of the curve (marked by **X**) at **P**, and **P** is consequently a farpoint of **C**. Thus, **Q** is closer to **C**, though it is further from either **A** or **B**.

point could also be a *farpoint* if the test point is outside of the positive radius of curvature of the surface in a concave region (see Fig. 4.15).

The algorithm has three stages:

1. A rejection test is applied to determine which mesh quads have possible Phong-nearpoints.
2. A quad tile is split diagonally into two triangles; this uses the same split as the code which converts  $(u, v, t)$  coordinates to points on the tessellated surface at a given subdivision depth (see Sec. 6.2). Under subdivision, as tiles shrink in size and become nearly planar, this algorithm will converge on a normal-nearpoint for the limit-surface. Generally, ambiguities which may arise due to splitting of the non-planar quads will disappear at the next level of subdivision.
3. The nearpoint on the surface is approximated by finding a  $(u, v, t)$  parameterized surface location which has a Phong-interpolated normal intersecting the test-point. That is, given a triangular patch with known positions, normals, and  $(u, v, t)$ -coordinates at the corners, positions and normals are linearly interpolated across the face of the triangle and are used to solve for a location with a normal that passes through the test-point (see Fig 4.16).

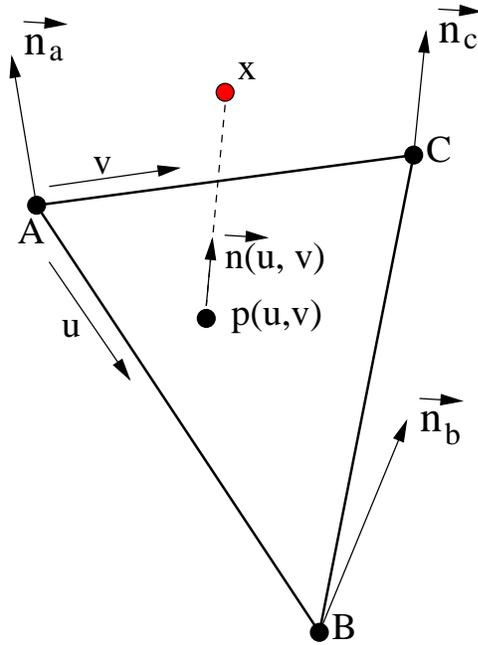


Figure 4.16: Finding a Phong-normal interpolant intersecting a position  $\mathbf{x}$ .

The rejection test uses a planar-sidedness test based on vertex positions and normals for quad or triangular faces. After its initial use to narrow the search space, the rejection test is used as the basis for a recursive, bilinear bisection search to locate the Phong-normal nearpoint. The nearpoint can be described based on the following mathematical derivation. Given the geometry as in Fig. 4.16, a simple bilinear interpolation produces the following parametric functions:

$$\begin{aligned}
 \mathbf{p}(u, v) &= (\mathbf{A}(1 - u) + \mathbf{B}u)(1 - v) + (\mathbf{C}(1 - u) + \mathbf{B}u)v, \\
 &= \mathbf{A} + (\mathbf{B} - \mathbf{A})u + (\mathbf{C} - \mathbf{A})(v - uv) \\
 \vec{n}(u, v) &= (\vec{n}_A(1 - u) + \vec{n}_{Bu})(1 - v) + (\vec{n}_C(1 - u) + \vec{n}_{Bu})v \\
 &= \vec{n}_A + (\vec{n}_B - \vec{n}_A)u + (\vec{n}_C - \vec{n}_A)(v - uv) \\
 \vec{m}(u, v) &= \mathbf{x} - \mathbf{p}(u, v).
 \end{aligned}$$

From these, the test condition follows: the point  $\mathbf{p}(u, v)$  is the local boundary point “nearest” (in the Phong-normal sense) the test-point if  $\vec{m}$  is collinear with  $\vec{n}$ , or trivially, if  $|\vec{m}| = 0$ . The problem can be reduced to one of finding a zero of a squared sine function, avoiding any divisions for computational efficiency, and it can be further

reduced by trigonometric identities to solving for the zeroes of

$$\begin{aligned} T &= ((\mathbf{x} - \mathbf{p}) \cdot (\mathbf{x} - \mathbf{p}))(\vec{n} \cdot \vec{n}) - [(\mathbf{x} - \mathbf{p}) \cdot \vec{n}]^2 \\ &= (\vec{m} \cdot \vec{m})(\vec{n} \cdot \vec{n}) - (\vec{m} \cdot \vec{n})^2. \end{aligned} \tag{4.15}$$

A bisection search was decided on after implementations that used a conjugate-gradient method proved too time-consuming for use in implicit surface routines. The bisection method has the additional advantage of interpolating the  $(u, v, t)$  coordinates across the tiles at the same time as it solves for the zero of the point-to-normal-vector distance function.

### 4.2.2 Drawbacks, extensions and improvements

Phong-normal nearpoints have several drawbacks. One is that test points must lie within a not-especially-well-defined “Phong-normal radius of curvature;” problems thus arise when surfaces are too highly curved for a coarse tessellation, due to the nature of Phong interpolation; wildly divergent normals can also lead to numerical errors on sidedness tests. Another problem is that the Phong-normal test is a strictly local measure; there can be numerous locations on a surface with a normal vector intersecting a given point in space, and the methods developed do not attempt to find these multiple points/regions and impose a Euclidean distance ordering on them. A nearpoint according to this definition may be a farpoint as well, again depending on implied boundary curvature. The Phong-normal technique, with various heuristics for resolution of ambiguities, has been suitable for implicit surface blending algorithms and for interactive techniques for selecting surface regions. It is, however, neither robust nor accurate enough for general use as a true subdivision-surface nearpoint estimator.

There are at least two ways its accuracy might be improved.

1. Use a higher order approximation to the correct normal, rather than the linear Phong approximation. This could be done as per van Overveld and Wyvill [151]; the authors also give a good discussion of the philosophy behind and drawbacks of Phong normal interpolation both in theory and in common implementations. Applying their ideas to the normal interpolations used in my nearpoint tests might be a fruitful avenue for future research.
2. Use a subdivision surface interpolation method, as developed by Stam [137],

to interpolate patch locations and normals directly. This is the ideal solution; from a coarse mesh, one can compute a correct solution for a nearpoint on the limit surface.

Of these two, the second is more exciting; since Stam's interpolation techniques apply to other stationary subdivision methods as well, this would give a generic, accurate nearpoint test for all such surfaces. For a Catmull-Clark surface, computational costs would be the one-time set-up of the b-spline patches for quads with normal 16-neighborhoods, plus additional one-time costs for quads with an isolated extraordinary vertex. The evaluation cost is thereafter the cost of a b-spline evaluation; it has been observed shown that such matrix-operations can be performed more than  $10^6$  times per second using Intel Pentium III hardware, ca. 2000 A.D.<sup>4</sup> In either case, using a non-linear interpolation will invalidate the recursive, bilinear subdivision search, with its simple accept/reject-test, but the conjugate gradient approach might still be workable.

Another area for improvement is the selection of a near-tile to begin the search, which currently uses the rejection test over all tiles at the particular subdivision level. There is a huge body of literature on proximity testing in computational geometry and simulation, including very rapid techniques involving spatial partitioning. Such global methods will be needed to solve the problem of choosing between multiple nearpoints, in any case, as necessary for an accurate, robust subdivision nearpoint algorithm.

### 4.3 Interpolating the medial structure of subdivision solids

The result shown in Sec. 4.2—that for a selected point in space (within curvature-defined limits), a near-point on an m-rep subdivision boundary can be computed—means that there are now tools for exploring the medial structure for IIS-surfaces. For a point on the boundary, a simple root-finding method can be used to find corresponding medial and involute positions to give a correct medial atom for that boundary point, based on the approximating Phong-normals to the surface. The algorithm is described below; Fig. 4.17 gives a graphical depiction. This method is similar to one

---

<sup>4</sup>Timings were by Paul Yushkevich at UNC-Chapel Hill, and used the Intel SML Small Matrix Library.

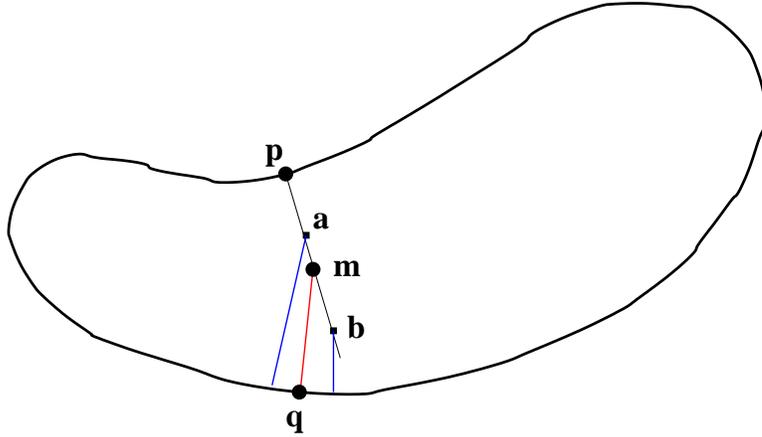


Figure 4.17: Finding an interpolating medial atom using bisection search along a Phong-normal. Using the normal from point  $\mathbf{p}$ , the nearpoint-function is used to find points on the opposing side nearest the bracketing points  $\mathbf{a}$  and  $\mathbf{b}$  along the normal. A bisection search is then performed on the segment between  $\mathbf{a}$  and  $\mathbf{b}$  to locate the point  $\mathbf{m}$  an equal distance from  $\mathbf{p}$  and a nearpoint  $\mathbf{q}$  for  $\mathbf{m}$  on the opposite side.

used by Yushkevich to compute medial atoms for involute positions on spline-based boundaries [164]. Thus, Phong-normals can be used to define a medial correspondence between boundary points; in the limit as the subdivision level increases, this approaches the correct Blum medial correspondence. Since this correspondence can be established at any level of subdivision, it creates a *Phong*-medial shape description for a tessellated surface.

### 4.3.1 Algorithm and implementation

The implementation is straightforward, and reduces to a  $1D$  rootfinding algorithm.

1. Select a point  $\mathbf{y}_1 = \mathcal{B}(u_1, v_1, 1)$  on the boundary of an IIS-surface of an m-rep.
2. Compute the Phong-normal  $\vec{n}$  at this point, and bracket a search interval

$$\mathbf{I} = [ \mathbf{y}_1 - (r - \epsilon)\vec{n}, \mathbf{y}_1 - (r + \epsilon)\vec{n} ],$$

where  $r$  is the approximate medial radius.

3. Perform a  $1D$  bisection search within this interval to find a point

$$\mathbf{x} \in \mathbf{I} : \exists \mathbf{y}_2 = \mathcal{B}(u_2, v_2, -1) \ni: |\mathbf{y}_1 - \mathbf{x}| = |\mathbf{y}_2 - \mathbf{x}|.$$

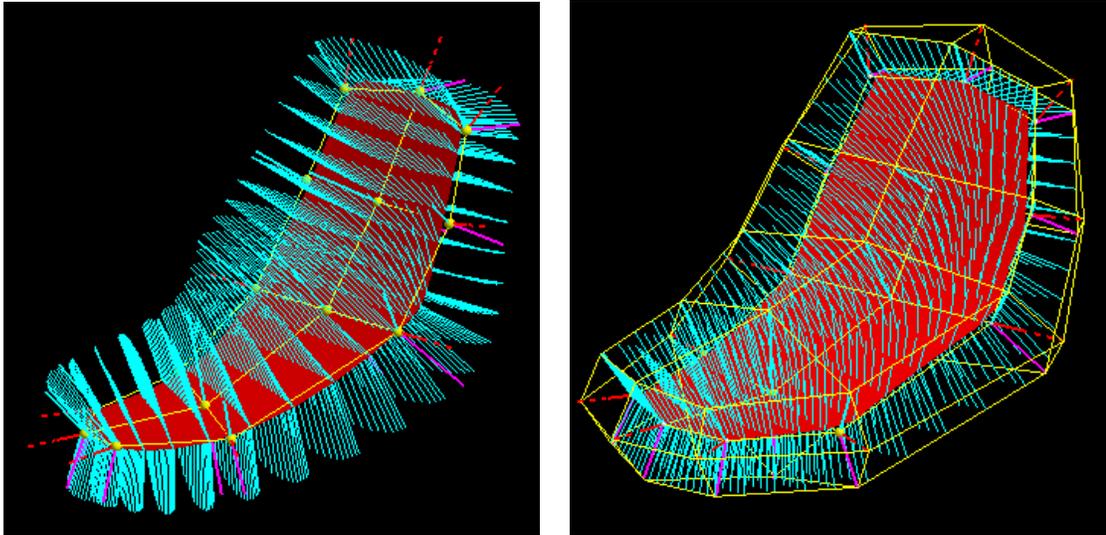


Figure 4.18: An m-rep resampled by Phong-normal medial interpolation. Note the similarity with the  $(u, v, t)$ -correspondence-based medial axis as shown in Fig. 3.18.

At known involute positions, given normal-interpolating IIS-surfaces, the computed medial locations should match the known values, subject to multiplicity problems. While this method runs into trouble along the crests and in other regions of high curvature, it is adequate to allow resampling of the medial mesh based on non-integer  $(u, v)$  values for the surface. Fig. 4.18 shows an example of this.

### 4.3.2 Discussion of Phong-normal medial axis

Given the  $(u, v, t)$ -correspondence-based medial interpolation as defined in the previous chapter in Sec. 3.3, the Phong-medial method is perhaps superfluous. The Phong-medial method has several drawbacks:

- computational intensity—while the bisection search is fast, the nearpoint code is still a bottleneck;
- philosophical dogma—it is too close in essence to trying to find a Blum medial point, an ill-conditioned task (and anathema to all right-thinkers);
- computational ambiguity—as a consequence of the last, it suffers from the same sorts of problems which plague other *intolerant* medial axis methods, including problems with branch-points and non-Blum regions.

The  $(u, v, t)$ -based method suffers from none of these drawbacks; it should therefore be preferred for modeling and m-rep-based analysis. The Phong-normal axis is still

of interest as a theoretical construct and perhaps offers another way to explore the geometry of objects defined by subdivision surface boundaries.

## 4.4 Concluding remarks on medially implied subdivision boundaries

This chapter described the creation and use of interpolating subdivision boundaries for m-reps, and it showed a vertex-regularization allowing IIS-surfaces to be constructed for all Catmull-Clark subdivision meshes. It also described an *as needed* normal interpolation for involute positions. Both the positional and normal interpolation add only  $\mathcal{O}(n)$  operations to the Catmull-Clark routines, where  $n$  is the initial number of mesh vertices and typically small. These methods can be applied as well to other stationary subdivision methods with known limit behavior, such as Loop subdivision surfaces.

A Phong-normal-based nearpoint proximity test was also described and discussed, as well as a Phong-normal-based medial axis, though this latter seems mainly of conceptual interest. The next chapter will present the final methods needed for multiscale m-rep modeling using subdivision boundaries—how boundary displacements may be mapped the subdivision surfaces, and how figures and subfigures may be attached either by implicit surface techniques or by a remeshing of their subdivision boundaries in the join regions.

# Chapter 5

## Object Modeling using M-rep Subdivision Solids

The previous chapter explained how interpolating subdivision surfaces are used to create boundaries for discrete m-reps. For m-reps to be broadly useful in shape modeling and shape analysis, two tasks remain to be explained:

1. the creation of fine-scale, tolerance-based surface deformation on subdivision surface boundaries;
2. the creation of multifigural models by smoothly joining additive and subtractive subfigures to a simple m-rep figure.

Section 5.1 discusses boundary deformation by two methods: displacement texture-mapping based on interpolated, medially based coordinates; and  $r$ -bounded displacement meshing for subdivision surfaces. It describes the prototype techniques developed for past and present m-rep modeling systems, places them in context with current work on subdivision surfaces and multiresolution meshes, and considers issues of hierarchical object simplification. Section 5.2 discusses how multifigure models can be created with m-reps. It describes the use of medial coordinates for the placement and joining of subfigures at figural boundaries, and it discusses the remeshing of interpolating subdivision boundaries to produce smooth joins between figures and subfigures, replacing the implicit surface techniques used for blending in previous applications.

## 5.1 Subdivision Boundary Displacement: Texture-based and Mesh-based Approaches

The medial-atom mesh of a discrete m-rep describes boundary positions to within an  $r$ -proportional tolerance; the information concerning fine-scale boundary details must be carried by the boundary itself. Thus, global shape characteristics are parameterized over variation in the medial skeleton, and local shape is parameterized over variation in the subdivision-meshed boundary. This is distinct from Fourier-based shape modeling methods provided by spherical harmonics, which lack locality in shape description, and distinct from wavelet and multiresolution mesh methods, which lack the global shape description and width-based tolerance given by multiscale medial methods.

For creating displacements on subdivision boundaries, computer graphics and CAGD offer two basic methods: displacement maps based on parameterized displacement images, and displacement meshes based on perturbations of subdivision mesh vertices. Displacement maps (as well as image maps) based on  $(u, v, t)$  medial parameterization will be discussed in the next section, and displacement meshes in the section thereafter.

### 5.1.1 Image and displacement textures for m-rep boundaries

The use of image textures to provide detail on parameterized surfaces dates back to Blinn and Newell [12], who based their work on the parametric rendering and texturing techniques developed by Catmull [27]. Texture-based surface displacements were first described by Cook in his work on shade trees [32], and displacement textures have been part of the Renderman API since the mid-eighties. The computer gaming industry has driven recent interest in them, seeking ways to increase scene detail without increasing polygon counts. A summary of this work is given by Doggett [46], who discusses both the history of displacement mapping and the state-of-the-art provided by current APIs and graphics hardware. Using current systems, it is now possible to do hardware-accelerated rendering of displacement-textured subdivision meshes.

The  $(u, v, t)$  coordinates of an m-rep boundary provide a straightforward parameterization for texture mapping applications. Prototype application of image and displacement maps to m-reps was implemented in *Rakshasa*, an early discrete m-rep

modeler that will be discussed in detail in Sec. 6.1. Rakshasa used the  $(u, v, t)$  coordinates of an m-rep figure as parameters for both image-texture mapping, via the OpenGL texture-mapping mechanism, and displacement mapping, via a simple vertex displacement scheme. The mapping to the image space is provided by simply scaling the  $(u, v)$  coordinates to  $(u/u_{\max}, v/v_{\max})$  and using these as texture coordinates for a  $[0, 1] \times [0, 1]$  image texture.

For displacement texturing, which was not supported by the OpenGL API or the graphics hardware of the day, subdivision mesh vertices at a given level are displaced in the vertex-normal direction according to a table lookup into a manually created displacement map where the displacement

$$\mathcal{D}(u, v) = \mathcal{F}\left(\mathcal{T}\left(\frac{u}{u_{\max}}, \frac{v}{v_{\max}}\right)\right)$$

is indexed by a surface position’s medially based  $(u, v, t)$  coordinates into a 2D scalar displacement image  $\mathcal{T}$  with coordinates in the range  $[0, 1] \times [0, 1]$ . The displacement image is stored as scalar intensities in the range  $[0, 255]$ , which the function  $\mathcal{F}$  then maps to the range  $[-\alpha, \alpha]$  for  $\alpha < 1$ . The new boundary position is then

$$\mathbf{x}' = \mathbf{x} + r\mathcal{D}(u, v)\vec{n}$$

where  $\mathbf{x}$  is the initial boundary location corresponding to  $(u, v, t)$ ,  $r$  is the interpolated medial radius, and  $\vec{n}$  is the surface normal. This implementation was a proof-of-concept only and ignored the value of  $t$ , thus holding the image values and displacements constant for  $-1 < t < 1$  along the crests. The value of  $\alpha$  gives a “collar” region within which displacements are kept. In Rakshasa,  $\alpha = 0.25$  was a constant, but a better scheme would be to vary it across the surface, so that

$$\alpha = \mathcal{A}(\mathcal{C}(u, v, t), \text{sign}(\mathcal{D}(u, v, t)))$$

would vary as a function of surface curvature measures (such as Gaussian or principal curvatures) and the direction of the displacement. Such a system would be useful to limit displacements into non-convex regions to avoid self-intersection of the surface.

Fig. 5.1 shows an image texture map and an associated, manually created displacement texture for a carved redwood plank, and Figs. 5.2 and 5.3 show the image and displacement maps as applied to a simple m-rep slab. Rakshasa did not compute perturbation of surface normals; these could be computed either from the perturbed

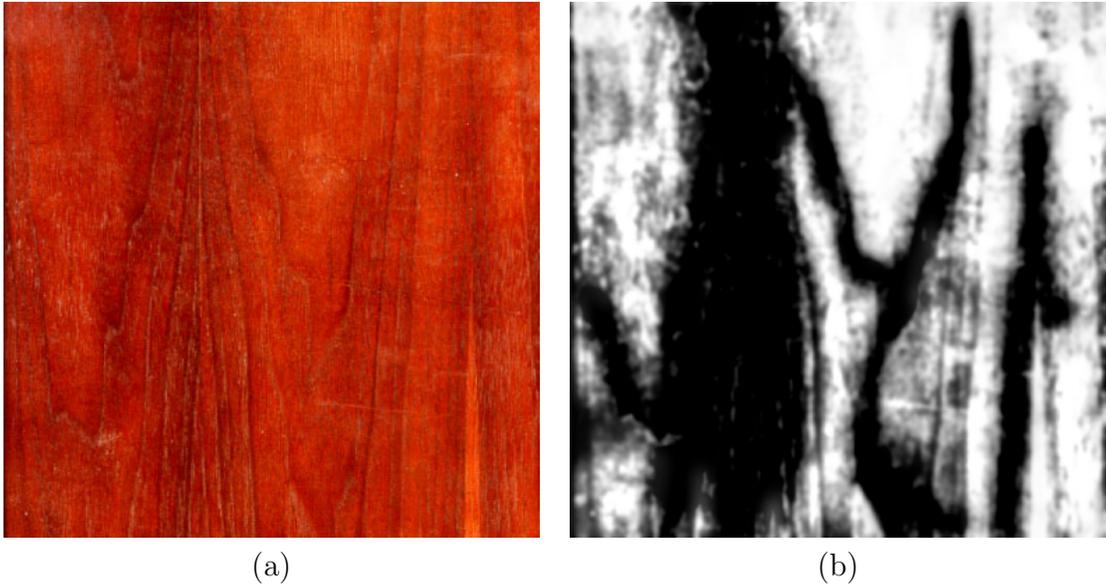
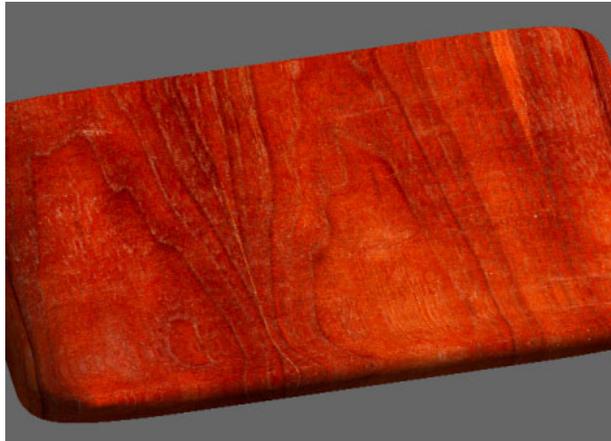


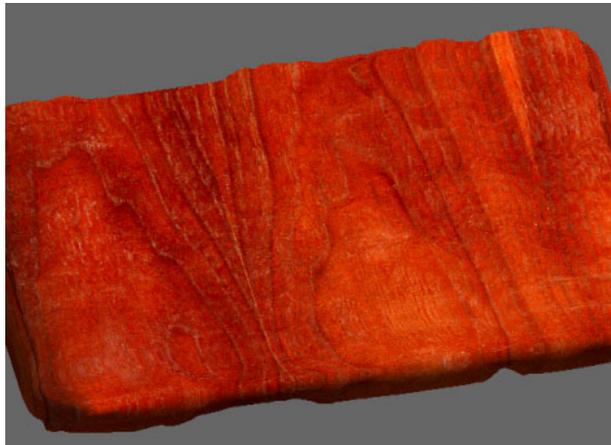
Figure 5.1: A redwood image-texture (a) and associated displacement map (b). The displacement map is a manually created scalar image where intensity corresponds to  $r$ -proportional displacement. The neutral grey value in this rendering represents zero displacement, the brighter, positive, and the darker, negative.

boundary mesh or by pullback functions from the boundary into the image  $\nabla\mathcal{T}$  of gradients of the displacement map itself. The implementation also ignored concerns regarding artifacts from point-sampling of the displacement maps. These would have been even more visible if normal-perturbations had been implemented. The expected problems are similar to aliasing-artifacts encountered when point-sampling height fields or disparity maps in image-based rendering.

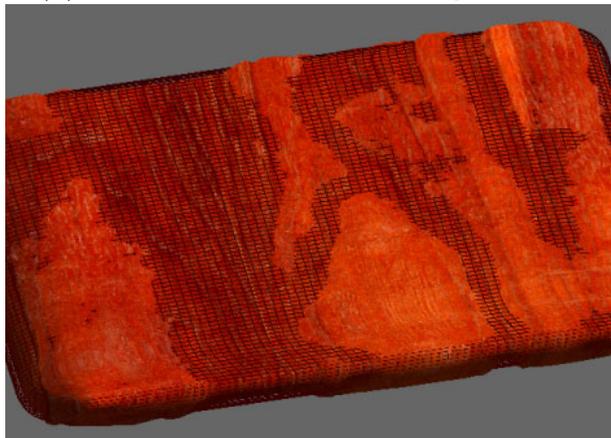
Displacement maps can be used as look-up tables for vertex perturbations of a subdivision mesh, as in these examples, but their advantages over displacement meshes (described in the next section) are only seen when using them in a procedural shading environment that allows displacement maps. In such an environment, the displacement textures can be applied to coarse-level subdivision surface polygons, avoiding the need to subdivide finely to get fine detail. They have a drawback of requiring separate maps for different levels of subdivision, at least between the coarsest levels and their successively refined meshes. At finer subdivision levels, the distances between corresponding positions on successive levels is small; thus, using the same displacement map might go unnoticed. Displacement maps might be derived for coarser levels by point-sampling based on coarse-level Phong normal intersections with the finer meshed surfaces. This could create a medially based LOD decomposition and



(a) undisplaced surface with image texture



(b) displaced surface with image texture



(c) displaced surface seen with the original, undisplaced boundary mesh

Figure 5.2: Image-based displacement mapping of an m-rep slab figure. All images use the same redwood image-texture. Image (b) and (c) apply displacements to the boundary according to the displacement map seen in Fig. 5.1(b). The m-rep  $(u, v, t)$  coordinates are used both for image-texture coordinates and as lookup values for vertex-displacements in the displacement texture.

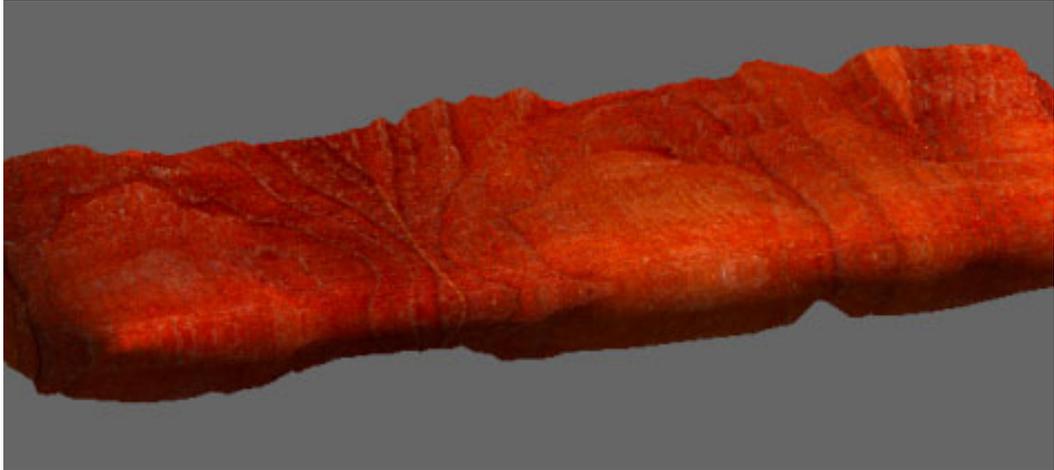


Figure 5.3: Edge view of the displaced m-rep boundary from Fig 5.2.

allow displacement textures to be substituted for highly subdivided surface geometry. This will be discussed later in the next section.

### 5.1.2 Displacement meshes for m-rep subdivision boundaries

Given a mesh-based boundary for an object, a *displacement mesh* stores a scalar or vector displacement at each mesh vertex. Displacement meshes have become increasingly popular in recent years, as seen in the previously discussed work of Lee [96] and Guskov [76]. Such techniques are well-suited to subdivision surfaces, forming the basis for much multiresolution mesh research, as discussed in Biermann [8], and they are especially applicable to m-reps, which are based on *interpolating* subdivision surfaces. Since there is no need to compute equivalent mesh perturbations between corresponding vertices at different subdivision levels, it is unnecessary to resort to multiresolution mesh editing techniques used with traditional Loop or Catmull-Clark surfaces. Perturbations at a vertex can be kept constant when finer-scale mesh vertices are interpolated by the subdivision process.

Perturbations defined at a coarse level can be interpolated using a number of different techniques: (1) the same splitting formulae as for the vertices themselves, (2) the simple bisection scheme used for the  $(u, v, t)$  coordinate interpolation, or (3) the same inversion method used to produce the interpolating subdivision mesh vertices. Method (1) will produce a Gaussian smoothing of the displacements, in the limit giving an almost-everywhere  $\mathcal{C}^2$  displacement field over the surface. Method (2), keeping coarse-level displacements constant, is more useful when precise perturbations

are defined at the coarse level and approximate perturbations are desired at a finer one. If the task is fitting an m-rep to image data, these approximate positions might be used as initial locations for a boundary search at the new vertices. Method (3) can be used to produce a continuous displacement field over the limit surface that interpolates the displacements at the coarse-level vertices; however, this field may have undesirable ripples—see comments below in Sec. 5.1.3 on  $r$ -field interpolation.

Fig. 5.4 shows an example of an m-rep model with displacement-mesh perturbations. This texture perturbs mesh vertices in their normal direction using Perlin noise (see Perlin [112]) modulated by the local  $r$ -value at each vertex. In other experiments, such  $r$ -based noise functions allow a scale-invariant perturbation of objects based on their local width function.

A perturbed subdivision mesh also needs new normals, both for rendering purposes and for nearpoint and other computations. Normals can be derived from the perturbed local grid structure by vertex averaging techniques or by more elaborate schemes, such as those of Desbrun et al. [44], computing differential geometric attributes such as normal vectors and curvatures on discrete grids representing a smooth underlying geometry. Perturbed normals computed on the finest-level mesh can then be substituted for those at corresponding coarse-level vertices, allowing Phong shading to approximate the finer-scale surface.

Normals can be computed on the fly when a model is deformed and then stored for rendering and computational purposes. They might instead be computed once and stored as rotations of the vertex normal; when the model is deformed medially, the same rotation would be used relative to the vertex’s local coordinate system. This would require a local frame at each vertex; this could simply be an interpolated  $q \in \mathbb{Q}$  frame from the medial frames associated with the initializing  $(u, v, t)$  involute positions. The quaternion frames could be interpolated by the subdivision just as the  $r$  values. This would be trivial to implement, but might not be worth the additional computational cost.

Displaced subdivision meshes are amenable to various LOD techniques for reducing polygon counts in rendering applications. The simplest is to use the coarser-level subdivision meshes with their displacements in place of the finer level meshes. While this will not prevent “popping” artifacts during substitution of coarser for finer meshes, it will be adequate for many purposes. More elaborate mesh simplification techniques, such as the simplification envelopes of Cohen et al. [31], are as applicable to the fine-scale displacement-meshes as they would be to any other polyhedral mesh.

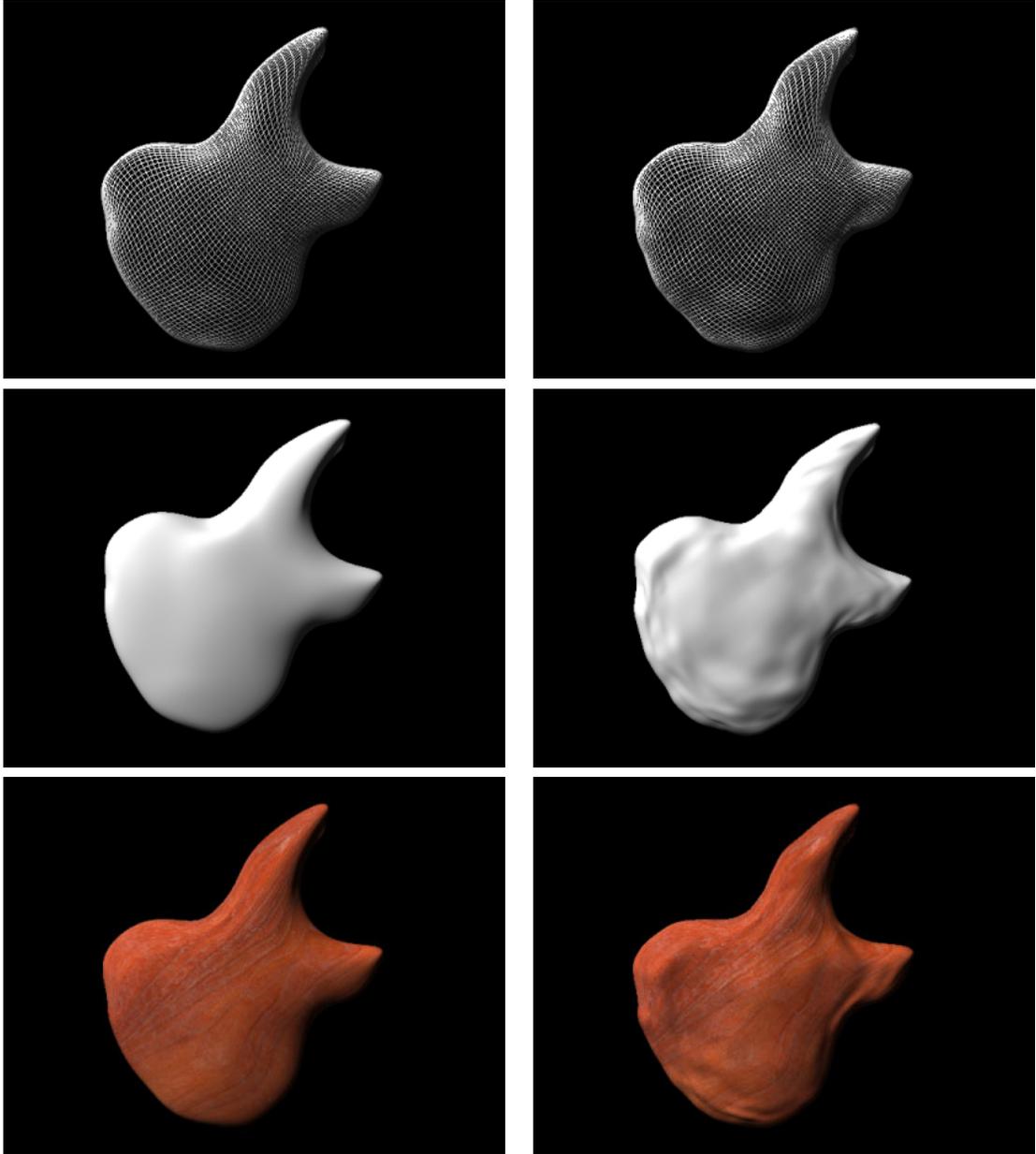


Figure 5.4: M-rep textured with displacement meshes. On the left is the unperturbed model; on the right is the perturbed model.

Other LOD methods involve the substitution of displacement textures on coarse-level meshes for the geometry at the fine-level meshes. These techniques can utilize the  $(u, v, t)$  parametric correspondence between subdivision levels in various ways. There are a number of interesting approaches:

1. creation of a bump map to perturb surface normals on polygonal faces to approximate the fine-scale curvature, matching normals at  $(u, v, t)$  locations in the fine-level meshes with their interpolated  $(u, v, t)$  positions on the coarser polygonal faces;
2. creation of a vector displacement map, according to the same  $(u, v, t)$  correspondence, giving the actual boundary displacement between coarse mesh-polygons and associated positions on the limit surface;
3. creation of a scalar displacement map, which would displace positions on coarse polygons along their Phong normals to positions on the limit surface; this would require computationally expensive ray-to-subdivision-surface intersection computations but could be done as a preprocess on a static model.

The latter two methods would need a bump map, as well, to specify normals at the displaced points. Methods (2) and (3) would only be useful in a procedural shading environment that allows displacement textures, whereas Method (1) would be applicable in standard rendering APIs such as OpenGL or DirectX. Ideas for more complex, scale-based LOD methods will be discussed in Sec. 7.1.

### 5.1.3 Modeling using tolerance-based boundary displacement

This section discusses conceptual notions necessary for using displaced m-reps effectively and efficiently for CAD and other modeling applications. The discussion, in general, will apply equally to texture-based and mesh-based boundary displacements and when this is not the case, it will state which method is applicable. Modeling using displacements has received attention in recent years, as in the work of Kobbelt [92] and Lee [96]. While the latter gives a good discussion of displacement-meshed subdivision surfaces, use of such surfaces for m-rep boundaries raises issues unique to medial, tolerance-based modeling systems.

## Significance of the interpolated $r$ -field on the boundary

Section 4.1.6 discussed interpolation of  $r$  and other geometric and medially implied values on subdivision boundaries. This is particularly necessary in tolerance-based boundary deformation, where the displacements are given as fractions of the  $r$  values at the positions on the boundary. Thus, for a point  $\mathbf{x}$ , each displacement value  $\delta(\mathbf{x})$  in the direction  $\vec{n}_{\mathbf{x}}$  is scaled by  $r_{\mathbf{x}}$  to give the actual boundary displacement.

It remains to be seen how accurate the interpolating  $r$  values need to be. As previously stated, running Catmull-Clark splitting on the  $r$ -values of the base boundary mesh does not suffice since this produces drift of correct values away from their known values at interpolated atom positions. One can perform a bilinear interpolation of  $r$  values, as is done for  $(u, v, t)$  values—this is currently done in Pablo—but this method fails to give proper  $r$  interpolation near bulges and other regions where  $\nabla r$  is significant. As possible improvements there are two methods: (1) creation of a continuous interpolating  $r$ -field on the subdivision limit surface using the same inverse method as that for the interpolating subdivision surface; (2) creation of an  $r$  interpolation across the medial surface, using a spline-based medial interpolation method or one of the better medial interpolation methods under development (see Sec. 7.1) and then using  $(u, v, t)$  correspondence to transfer these  $r$  values to the boundary. Both of these methods may have problems with unconstrained interpolated  $r$ -fields showing too much wiggling; an *accurate* medial interpolation would avoid this problem but would be ill-defined in non-Blum medial regions—any medially based  $r$  interpolation will fail for surface points in non-Blum regions, and the medial behavior in blend regions is problematic.

The dependence of the boundary displacement on  $r$  implies that standard displacement textures cannot be used interactively in procedural displacement shading unless the displacements can be made a function of both the interpolated  $r$  values and displacement  $\delta(\mathbf{x})$  fields. Thus, if an object is scaled along a length axis to taper off, the displacements must automatically taper off as well.

## Dynamically editing boundary displacements

Specifics of current m-rep modeling systems will be discussed in Ch. 6. For general m-rep modeling systems, however, there are considerations needed for object modeling with boundary tolerance that will be discussed here.

It is fundamental to m-rep modeling that boundary displacements be limited to be

within width-proportional tolerances, to maintain the separation between large-scale medial structure—modeled by the figure-subfigure hierarchies—and fine-scale boundary detail—modeled by displacement meshing or texturing. Thus, any system for editing boundary displacements must inform the user when a displacement exceeds a parameter giving the maximum fraction of  $r$  allowed for a boundary displacement. It is not necessary to restrict changes to the medial  $r$ ; since the boundary displacements are given in units of  $r$ , changes to medial  $r$  will affect displacements in corresponding boundary regions in direct proportion. A modeling system should be able to substitute a joined subfigure for a displacement that exceeds its limits.

It is also necessary to limit displacements based inversely on boundary curvature to prevent boundary self-intersections. In order to do this dynamically during interactive editing, both boundary-displacement and medial-shape changes are relevant, and both need to be restricted in accordance with approximate mesh-curvature metrics. This and other operations will be discussed in Section 7.3.2.

Various tools are desirable for displacement editing. Polyhedral sculpting templates can be applied to an m-rep, with its boundary mesh perturbed in normal directions to intersect the templates, as was done in Fig. 5.4. Displacements could also be specified interactively using a Gaussian weighting over a region selected on the boundary. For a displacement mesh, this would be applied to mesh vertices in  $\mathbb{R}^3$ ; for a displacement texture, the Gaussian weighted displacement would be achieved by a pullback from the boundary point into the texture-space. Again, if such a displacement exceeds  $r$ -proportional limits, the modeling system would substitute a joined subfigure in place of the the displacement.

## 5.2 Multifigure Modeling using M-rep Subdivision Solids (co-authored with Qiong Han)

Multifigure modeling is inherent in the nature of m-reps, which deliberately reject the instability of the branching medial structures of tradition skeletal methods. Complex m-reps, therefore, are created by directed acyclic graphs (*DAGs*) of figures and subfigures. These DAGs are often further restricted to hierarchical trees with parent and child figures; this discussion will use *parent* and *child* interchangeably with *figure* and *subfigure*, though the discussion refers equally to trees and more general DAGs. Research on multifigure m-reps has developed two key techniques:

1. the use of medially implied coordinates for positioning subfigures at figural boundaries;
2. the use of subdivision surface remeshing to achieve smoothly blended joins between figure and subfigure, allowing modeling to be approached using a CSG paradigm with additive and subtractive subfigures.

Initial m-rep experiments by Thall (for the Rakshasa application) linked subfigures to figures by tying them to medial locations—first to discrete atoms in the medial mesh and later to  $(u, v)$  positions on an interpolated medial sheet. With the abandonment of an explicit medial interpolation, it was necessary to find other ways to specify figure-subfigure relationships. The plan finally implemented was based on the  $(u, v, t)$  boundary coordinates described in Ch. 3, and early implementation experiments and development in Pablo were done by P. Thomas Fletcher. Meanwhile, work on joining figures and subfigures proceeded in several different directions but culminated in the work of Thall and Qiong Han, who demonstrated the first working scheme for subfigure attachment by remeshing of subdivision boundaries (see Han et al. [78]). These techniques will be discussed in turn below.

### 5.2.1 Medial coordinates for subfigure positioning

The Pablo development team chose to link subfigures at positions where they have discrete medial atoms near  $(u, v, t)$  specified locations on the parent boundary. These *hinge atoms* align a row or column of the subfigure medial sheet with the parent figure boundary and allow the subfigure to be rotated transverse to the hinge direction (hence the name). Fig. 5.5, a closeup of an indentation figure’s medial mesh, shows the hinge atoms at their anchoring positions on the parent figure. This method for attaching figures and subfigures can be separated into two somewhat orthogonal decisions:

1. binding subfigures to the medial geometry of the parent;
2. placing subfigures at boundary  $(u, v, t)$  rather than medial  $(u, v)$  locations on the parent figure.

#### Binding subfigures to the medial geometry of the parent

A subfigure is a part of an m-rep figure. As such, a child figure needs to be integrated into the medial geometry of its parent for two reasons. First, transformations of the

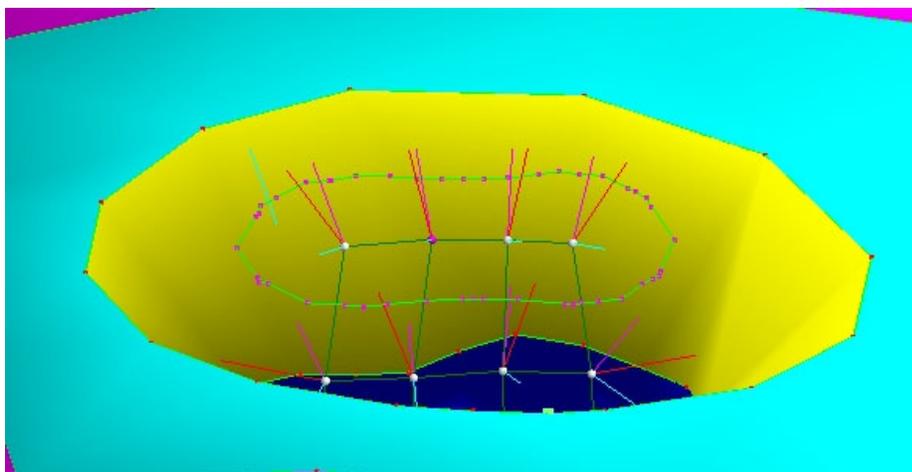


Figure 5.5: An indentation figure with visible hinge atoms on the (invisible) parent figure boundary. The inner green polygon shows the intersection of the original subfigure with the parent figure boundary prior to the blend.

whole subfigure can be specified relative to the parent. Second, changes to medial structure of the parent can be reflected in the child as well. By positioning a subfigure’s medial hinge-atoms using the medially implied coordinates of the main figure, both of these interactions are facilitated; subfigures may be modified by specifying changes relative to the parent geometry, and deformations on the subfigure can be computed in correspondence to deformations of the parent.

Modifying a subfigure relative to the parent medial geometry allows changes to the subfigure to be parameterized in parent-figure-based terms. This is important in segmentation based on deformable models, where a fixed figure may have variability in the characteristics of a subfigure—the figure can be kept in place and the subfigure transformed in parent-figure-local coordinates to effect an automatic fitting to the data. In modeling applications, parent-figure-coordinates provide a stable parameterization for positioning and manipulating subfigures while maintaining desired structural connectivity and overall object shape. In particular, the figural coordinates establish a correspondence that is invariant under similarity transformation as well as under medial and boundary deformation of the parent, as will be described below.

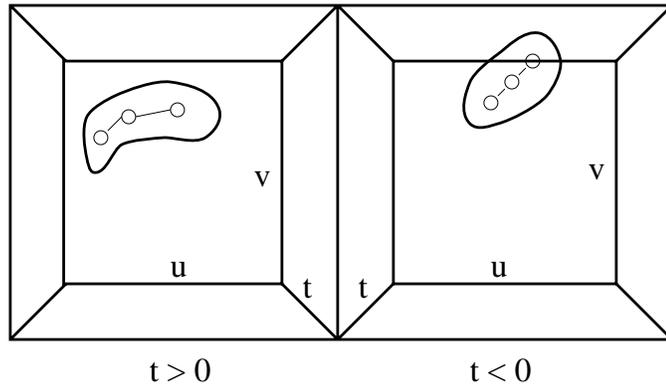
When a parent figure is deformed medially, the child figure must not simply maintain positional correspondence with the figural boundary but also bend and deform in ways corresponding to the desired behavior of a protrusion (or indentation) of the parent. For example, if a figure is bent or stretched about its medial axis, the hinge atoms of a subfigure will be translated and rotated to maintain their position and

orientation relative to the changed  $(x, y, z)$  positions and orientations of their  $(u, v, t)$  anchor points on the parent. Similarly, boundary displacements on a parent figure can be propagated to the child as well.

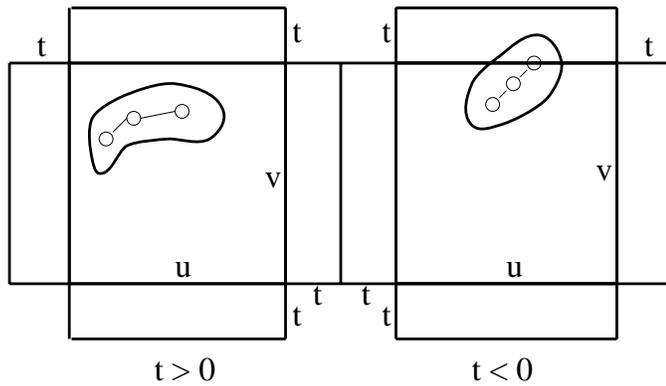
The changes to a figure in the neighborhood of a subfigure hinge can be propagated to the subfigure using the coordinate systems provided by the hinge atoms, and changes at these positions can be propagated along the subfigure’s medial mesh using various modeling-heuristics, such as a damping effect proportional to mesh-distance from the hinge. In this way, distortions of the parent figure will produce varying degrees of distortion of connected subfigures, according to individually tunable parameters. This approximates the behavior of an elastic solid under deformation and is especially necessary for on-going research on multifigure m-rep segmentation, where an m-rep must be automatically deformed to produce an optimal fit to 3D image data. Another approach to m-rep deformation uses finite-element methods (*FEM*) to model the m-rep and its subfigures and compute deformations using physically based models. See discussion of the work of J. Crouch [33, 35] in Sec. 6.3.3 on using m-reps and FEM to model tissue deformation.

### Placing subfigures at figural boundaries

Connection of subfigures at the boundary contrasts with the earlier attempts to link subfigures directly to medial locations on the parent figure. There are several reasons for making this choice. On theoretical grounds, linking boundary protrusions or indentations to the medial surface leads back to the small-scale intolerance and brittleness of Blum axis methods, where boundary perturbations produce discontinuous changes on the medial structure. Pragmatically, medial subfigure linkage was made difficult for m-reps by the lack of a good continuous interpolation of the medial surface—this was prior to the work of Yushkevich on cm-reps [163] and the most recent work of Kerckhove on interpolating medial surfaces for discrete m-reps [91]. The latter will be discussed in Sec. 7.1. Placing subfigures at parent figure boundaries also avoids the medial-to-boundary correspondence problems that arise with continuous medial representations and allows straightforward attachment of subfigures along the crest of the parent, where the 1-to-many nature of the medial- $(u, v)$ -to-boundary mapping makes subfigure attachment to the medial surface problematic. Fig. 5.6 illustrates the hinge atoms and boundary location intersections of an m-rep subfigure with a parent figure in the  $(u, v, t)$  space of the parent.



(a) A parametric map showing the connectivity of  $(u, v)$ ,  $(u, t)$  and  $(v, t)$  coordinate patches.



(b) A parametric map showing undistorted iso- $u$ , iso- $v$  and iso- $t$  lines on coordinate patches.

Figure 5.6: Placement of a subfigure in  $(u, v, t)$  parameter space of parent figure. In (a), the left and right edges are connected, and the top and bottom left-hand edges connect to the top and bottom right-hand edges respectively. In (b), the parametric map can be folded without warping, like a paper box, creating the 1-1 cover of the genus-0 (spherical) topology of an m-rep slab figure.

### 5.2.2 Smoothly joining figure and subfigure by remeshing

M-reps require subfigures to join figures in a smooth blend having the same curvature continuity as the rest of the m-rep boundary. This is in contrast with the sharp, tangent discontinuities of traditional CSG modeling and its extension to subdivision surfaces in work such as Biermann’s; such fine surface detail is carried by the m-rep boundary displacements and is not part of the medial figural structure.

Early work on figure-subfigure attachment was based on implicit surface blending techniques; indeed, this was an early aspect of m-reps as new geometric primitives, ideas developed by Thall (in discussions with Turner Whitted) to create medial primitives as extensions of blobby modeling and convolution surface primitives and eschewing any explicit surface representation. Early experimentation with implicit surface methods was discussed in Sec. 2.4.3 and culminated in the experiments of Fletcher [53]. This work was abandoned when subdivision surfaces were chosen as a more tractable boundary representation for fleshing a skeleton of medial atoms.

Early figure-subfigure attachment in Pablo was also based on implicit surface methods; in this case, with subdivision surfaces already in use as figural boundaries, a distance function over each boundary (computed using methods from Sec. 4.2) was used as the basis for a implicit surface blend between figure and subfigure. An isosurface marching algorithm was then used to construct the blended surface, with each point maintaining its  $(u, v, t)$  association with the initial figure and with a reparameterization in the blend region. This technique, implemented by Gash and Thall, with input from Pizer and Fletcher, served for early multifigure experimentation in Pablo but suffered from several drawbacks. Quirks in the nearpoint code led to problems in areas of high curvature—which, unfortunately, were exactly the blend regions. Also, the blending was typically too slow for interactive modeling, due to the cost of the distance function being evaluated on the subdivision boundaries, which involved a bracketed search at fine levels of subdivision.

A scheme to blend m-rep subdivision boundaries must overcome several difficulties. Problems particular to m-rep interpolating subdivision boundaries are in the figure-subfigure intersection computation and in the remeshing of interpolating subdivision meshes to produce a smooth blend region. In the case of m-reps, a sharp Boolean cut at a join is exactly what is not desired; fine-scale structure in the blend will be created by the same boundary displacement methods as in non-blend regions. As will be seen below, the techniques developed for m-reps allow the degree of curvature in a blend region to be adjusted by setting blending-parameters.

For figure-subfigure joining, m-reps do not require the full power of general intersection methods—discussed in Grinspun et al.[73]—as would be needed for collision detection and other physically based applications. In the figure-subfigure case, a subfigure to be joined is placed deliberately at the boundary of the parent figure with a selected row of hinge-atoms placed in proximity to the parent figure boundary.

The remeshing of subdivision surfaces in join regions has unique aspects due to the interpolating nature of m-rep subdivision boundaries. While it is the interpolating limit surface that must be joined, the actual remeshing uses the inverse mesh positions computed during the surface fitting stage described in Sec. 4.1. Once the join region has been determined, the parent and child meshes must then be stitched to form a closed surface mesh.

### 5.2.3 Joining figure and subfigure by remeshing: implementation

A prototype application for figure-subfigure joining of m-reps by remeshing of subdivision boundaries has been implemented by Qiong Han at UNC-Chapel Hill, based on ideas jointly developed with Thall and with input from Stephen Pizer, Sarang Joshi, and other members of the Pablo development team at MIDAG. The techniques are general enough for both protrusion and indentation figures, and they allow parametric control of the “sharpness” of the join in the blend regions. Fig. 5.7 shows examples of protrusion and indentation subfigures produced by the current application.

The algorithm works by intersecting the figure and subfigure at the coarsest subdivision level to get an approximate curve of intersection, producing a cut-curve on figure and subfigure, using the  $(u, v, t)$  coordinates of the intersection curve on the two figures. It then creates a new initial mesh by knitting the original initializer meshes by triangulating the region between the cut-curves. The algorithm proceeds in the following steps:

1. Calculate the intersection curve between the figure and subfigure meshes on the base-level interpolating surfaces. The intersection calculation uses the subdivision-surface proximity code described in Sec. 4.2. This generates an approximate intersection curve as a closed polygon in the parent figure’s  $(u, v, t)$  parameter space.
2. The intersection curve is subsampled in the  $(u, v, t)$  domain and then dilated in the parameter space of the parent figure. The new curve vertices are then

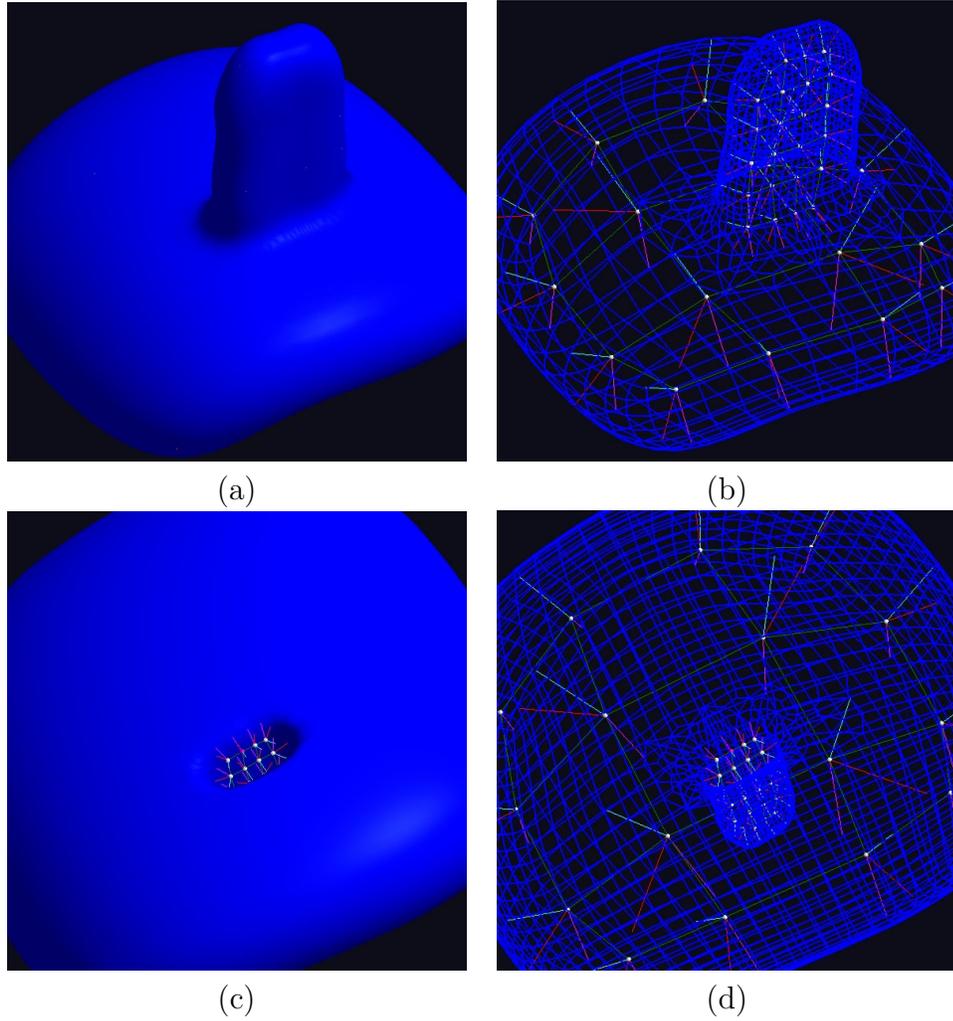


Figure 5.7: Examples of protrusion and indentation subfigures. Images (a) and (b) show a protrusion subfigure and Images (c) and (d), an indentation subfigure.

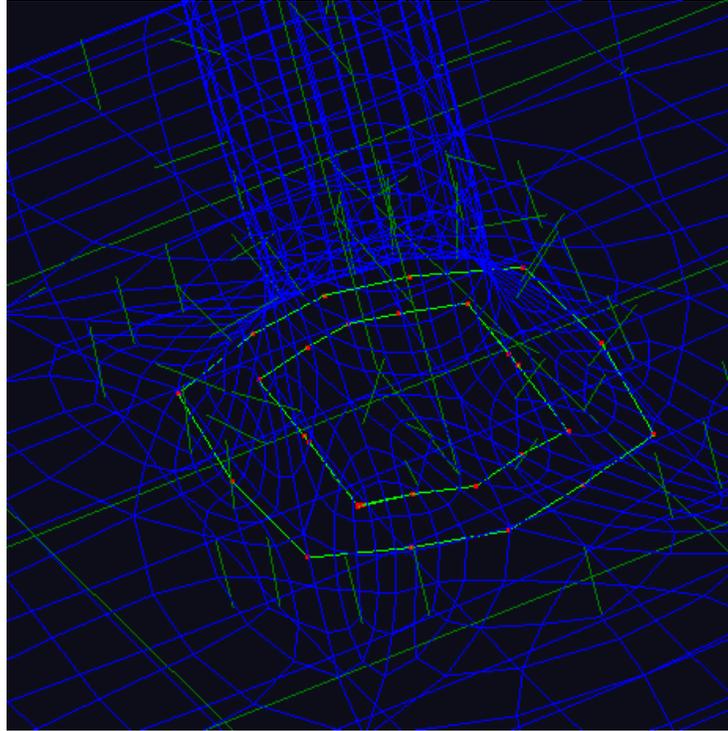


Figure 5.8: Figure-subfigure intersection curves. The images shows the blended subdivision surface (not at level 1) with the intersection curve and the dilated intersection curve from the level 1 mesh intersections.

mapped back to Euclidean space locations on the parent figure. This ensures that the dilated curve remains on the parent boundary. An example of an intersection curve before and after dilation can be seen in Fig. 5.8, and a view of several intersection curves and their dilations in the parameter space can be seen in Fig. 5.9.

3. Cut the subfigure by a pre-defined  $u$  (or  $v$ ) value, offset from the intersection curve, according to an offset parameter. The subfigure is sliced cleanly at a fixed  $u$  or  $v$  value—this is a current restriction of the software.
4. Cut the main-figure by the dilated intersection curve, eliminating vertices within the curve and triangulating the neighborhood of the cut as necessary.
5. Knit the figure and subfigure together by triangulating along the cut to produce a new initial (i.e., not limit surface) mesh for subdivision. This is now a hybrid mesh of both quads and triangles, which can be subdivided using the revised Catmull-Clark rules for meshes with non-quadrangular faces. The triangulation

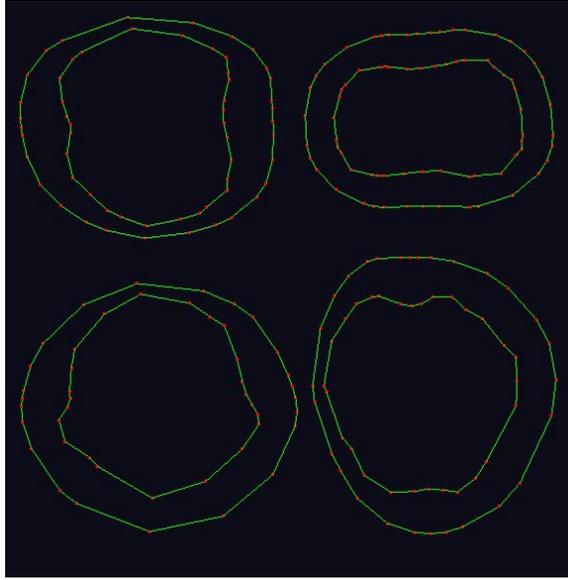


Figure 5.9: Dilation of intersection curves on parent figures. Sampled points on the intersection curve are displaced in  $(u, v, t)$  parameter space. The dilation is done in multiple small steps, varying in distance and direction according to the local approximated tangent and curvature of the surface-curve. This prevents self-intersection and produces a smoothing of the initial curve.

of the blend region uses simple heuristics to achieve an adequate but in no way optimal remeshing.

6. Subdivide once *in parameter space* to produce a level-2 mesh, and construct a new initializing mesh by placing each vertex at its  $(u, v, t)$ -corresponding  $(x, y, z)$  location on the original (unblended) figure or subfigure grid. Using the remeshed initial grid directly produces too many artifacts, due to the coarseness of the tessellated boundary. By tying the connectivity of the remeshed surface to the level-2 boundaries, deviation from the original interpolating surface is minimized and the changes in the blend region are more localized. In the blend region, there are no corresponding vertices on the unblended meshes; this is dealt with by simply removing the new vertices and using leaving the blend region triangulated (rather than quads, as is the rest of a level-2 mesh).
7. This new level-2 mesh is now the control mesh for the new interpolating subdivision surface, and smoothing Catmull-Clark subdivision will generate higher level meshes or points on the limit surface.

Fig. 5.10 shows the remeshing in the blend region for a protrusion or indentation fig-

ure. Our use of curve-dilation in  $(u, v, t)$  in the m-rep boundary remeshing scheme was developed independently but directly parallels the use of cutting curves in parametric domains as developed by Biermann et al. [8], as discussed in Sec. 2.5.4.

The behavior of the blend region can be controlled by two parameters, one which determines the amount of dilation of the cut on the parent figure, and the other which determines the offset for the cut, in  $u$  or  $v$ , from the intersection curve on the subfigure. Figs. 5.11 and 5.12 illustrate this for protrusion and indentation figures.

### 5.2.4 Fitting a polygonal model with a blended, multifigure m-rep

As a demonstration of the techniques developed for multifigure m-rep modeling, a polygonal model was fitted with a multifigure m-rep using the figure-subfigure blending described in this section. The polygonal model, `bigspider.ply`, has multiple overlapping polygonal meshes forming the body, legs and other features of a spider; it was obtained from the Computer Graphics Group at the University of Virginia.<sup>1</sup> The spider model has 9286 triangular faces meshing 4670 vertices (see Fig. 5.13).

Since code to fit m-reps directly to polygonal models has not yet been implemented, an m-rep model was created for `big_spider` in the following stages:

1. a 3D scan-conversion was done to create a voxelized, binary volume-image of `big_spider`;
2. the Pablo modeling system (described in Sec. 6.2.1) was used to manually create and position multiple figures for the spider body, legs and mandibles;
3. Pablo's automatic registration capabilities were used to refine the model, segmenting the binary image, using the hand-built multifigure m-rep as a initial template, and using subdivision-surface blending to join the multifigure object.

Displacement textures were not fit to the model, leaving an undetailed spider but one that illustrates well the subdivision surface blending. The final spider m-rep had 208 medial atoms—the cephalothorax and abdomen each had a  $4 \times 4$  medial mesh, the legs each had 20 medial atoms and each mandible had 8. The counts for the legs and mandibles are misleading—the actual medial sampling was 10 atoms per leg and 4 per mandible, but the lack of single-chain m-reps in Pablo required using paired,

---

<sup>1</sup><http://www.cs.virginia.edu/~gfx/Classes/2001/Advanced.Spring.01/plymodels>

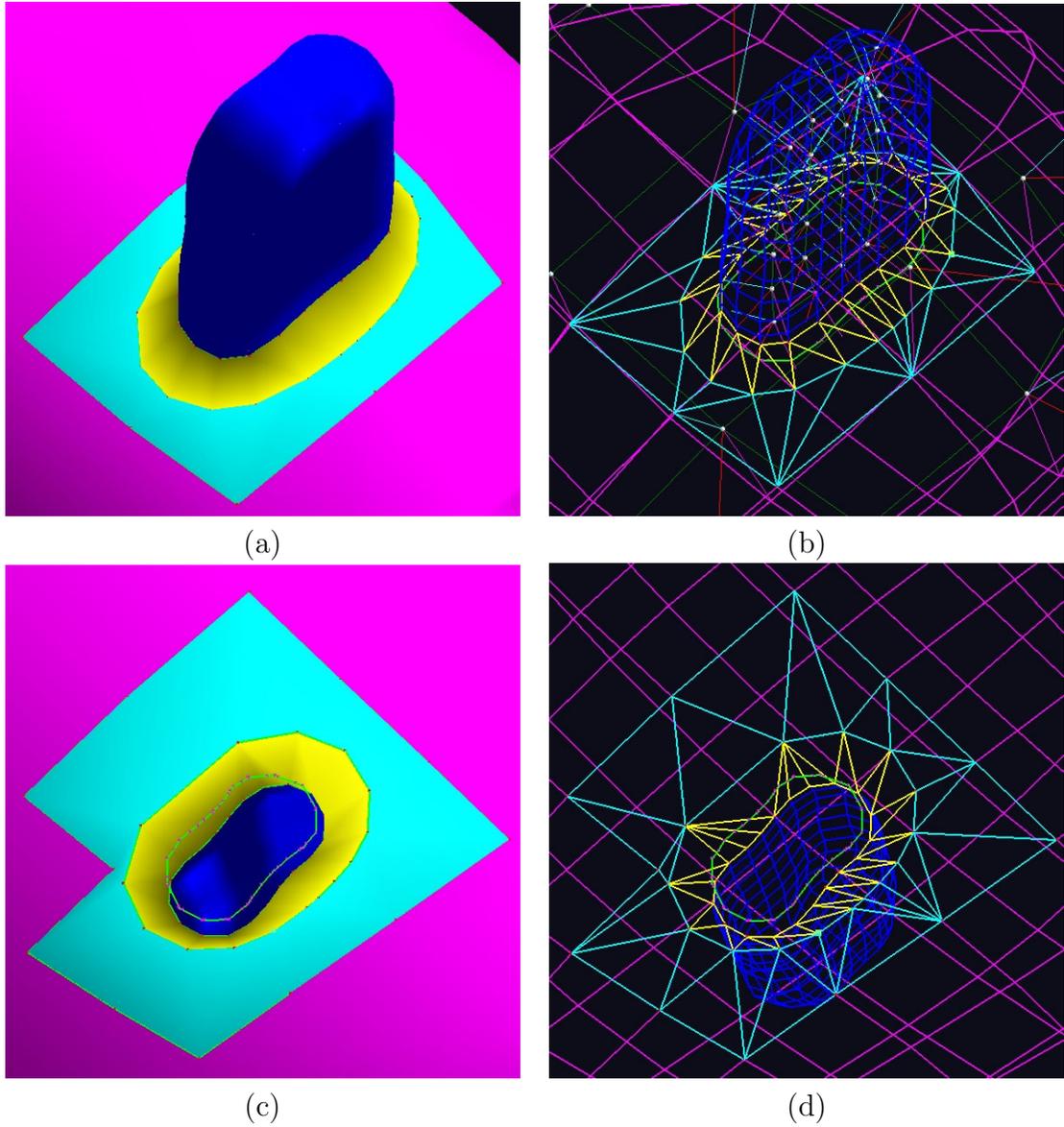


Figure 5.10: Detail of figure-subfigure remeshing in region of join. Images (a) and (b) show a protrusion subfigure and Images (c) and (d), an indentation subfigure. The cyan region is the remeshing of the parent figure. The yellow region shows the knitting of the remeshed parent and child figures at the join.

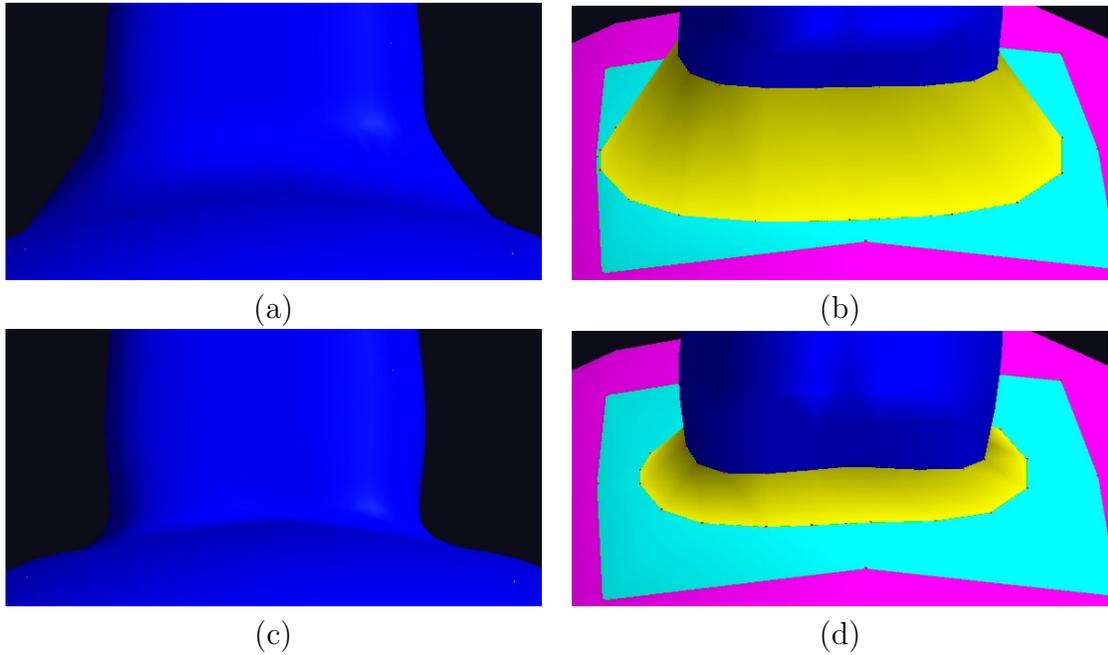


Figure 5.11: Adjustment of blending parameters for protrusion subfigure. Images (a) and (b) show a wide blend region based a large dilation on the parent figure and a large offset of the cut on the subfigure. Images (c) and (d) show a narrow blend regions, produced by a small dilation and small offset.

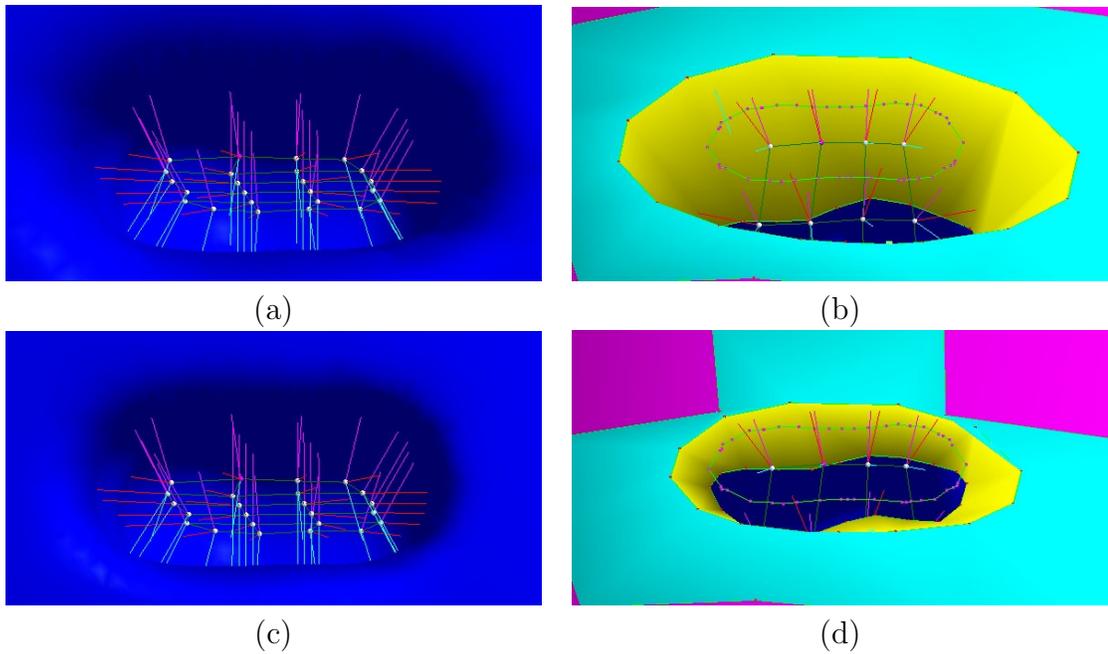


Figure 5.12: Adjustment of blending parameters for indentation subfigure. Images (a) and (b) show a wide blend region based a large dilation on the parent figure and a large offset of the cut on the subfigure. Images (c) and (d) show a narrow blend region, produced by a small dilation and small offset.



Figure 5.13: The `big_spider.ply` model from the Computer Graphics Group at the University of Virginia. This model has 9286 triangular faces meshing 4670 vertices.

doubled medial atoms in  $10 \times 2$  or  $4 \times 2$  grids for each leg or mandible respectively. Thus, the actual medial sampling for the model was 120 medial atoms. Fig. 5.14 illustrates the medial sampling and mesh structure for the spider m-rep. Subdivision boundaries applied to this m-rep are shown in Figs. 5.15 and 5.16. Close-up details of the figure-subfigure subdivision blends are shown in Figs. 5.17 and 5.18.

### 5.2.5 Implementation issues and drawbacks

In the current implementation, a single figure-subfigure blend takes on the order of 0.5 seconds to compute on an average PC. While this is considerably faster than implicit blending methods, it is still slow for interactive application. With straightforward optimizations, however, the blending of subdivision surfaces should be possible at interactive rates, allowing subfigures to be manipulated and the blended boundary recomputed on the fly.

Early implementations of Pablo were not able to place a subfigure at a corner of the medial medial sheet, covering crest regions with both non-constant  $u$  and non-constant  $v$  values, as in Fig. 5.19. This created problems for the dilation of the intersection curve, which must displace the curve based on local tangents and curvature in the

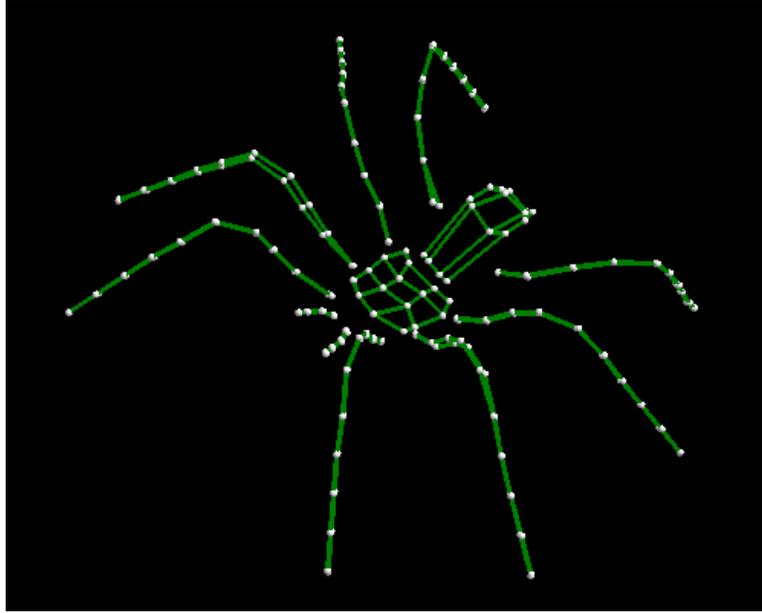
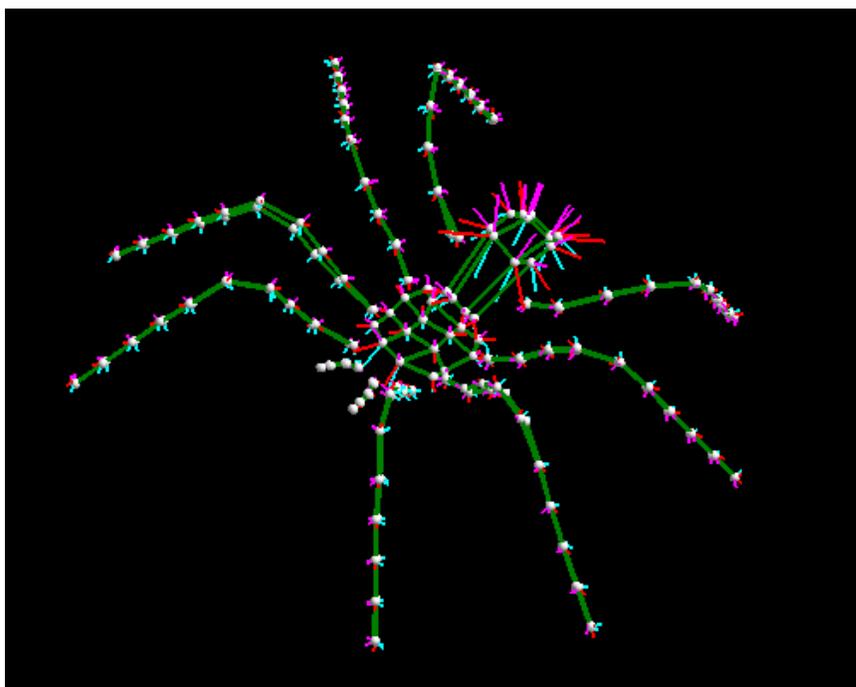


Figure 5.14: Medial structure of the spider m-rep. This model has 208 medial atoms, but the actual number of medial samples is only 120.

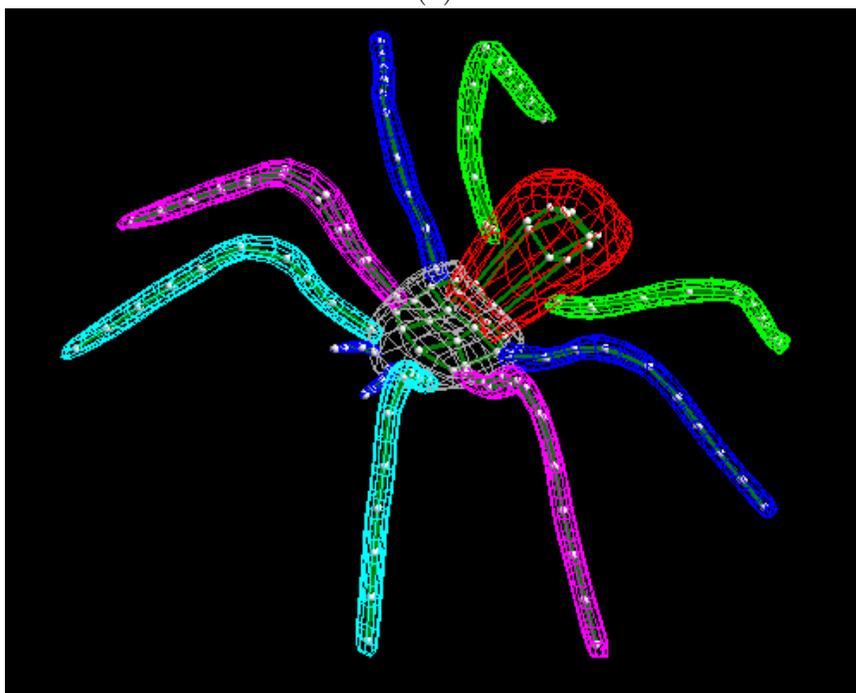
parametric domain. Problems arose from the skewing of coordinate axes in these regions—the  $u$  and  $v$  axes are, in fact, coincident along the diagonals. This was strictly a problem with the patch coordinates—the surface is  $\mathcal{C}^1$  continuous at the corners between patches. The solution of this problem involved reparameterizing these regions into local patch coordinates and then mapping back to  $(u, v, t)$ . Such a reparameterization was also done by Crouch in her m-rep FEM research [33].

There has been no discussion of reparameterization in blend regions, where the medially implied  $(u, v, t)$  coordinate system becomes ambiguous. For displacements based on displacement meshes (or other multiresolution mesh methods), there is in fact no need for reparameterization in the blend. For displacement texturing, a new texture map will be needed for the blend region and texture coordinates created for it. Issues of  $r$ -interpolation, necessary for displacement bounds, also remain to be addressed. It remains to be seen whether a straightforward interpolation of  $r$  by the current subdivision rules will produce acceptable approximations to an “accurate” value of  $r$  for displacements in the blend regions. The general behavior of displacements in the neighborhood of a blend and the interpolation of displacements across blend regions must also be explored.

Another implementation problem is how to attach a subfigure in a blend-region of an already multigure m-rep. This created problems for the spider model above;

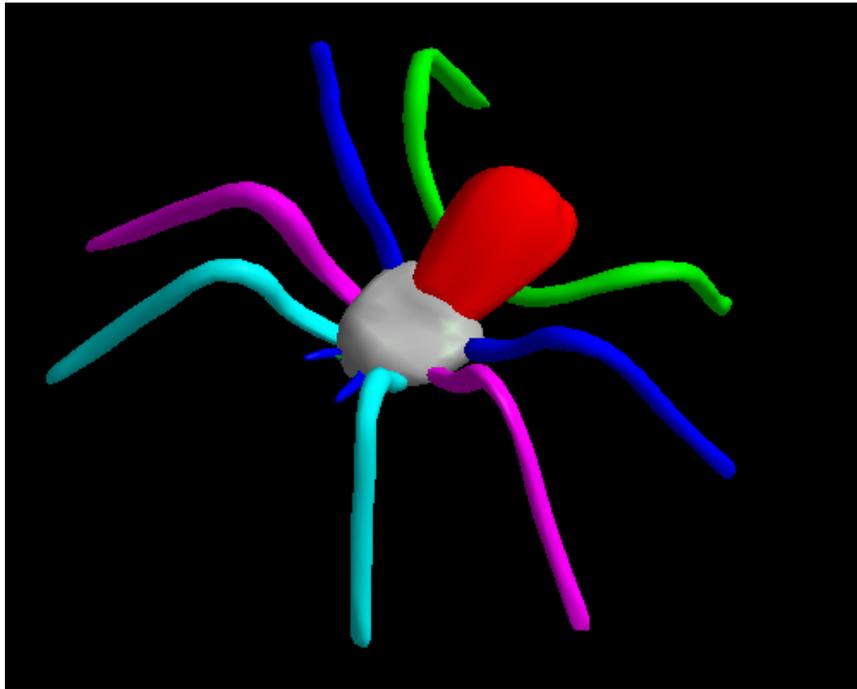


(a)

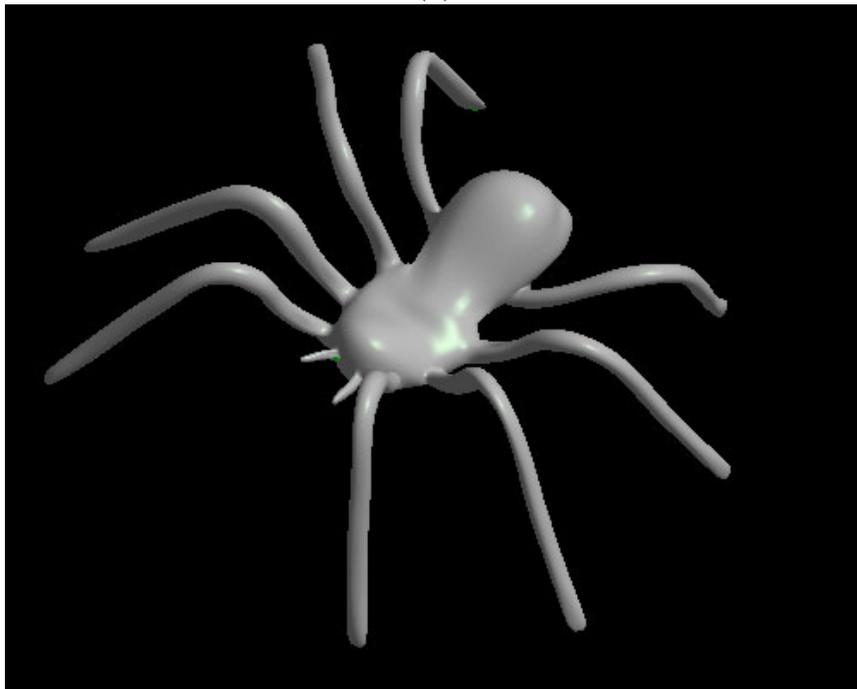


(b)

Figure 5.15: Subdivision boundaries for the spider m-rep; shown are (a) the medial mesh with atoms and involute vectors, (b) level-2 subdivision surfaces created individually for each figural mesh.

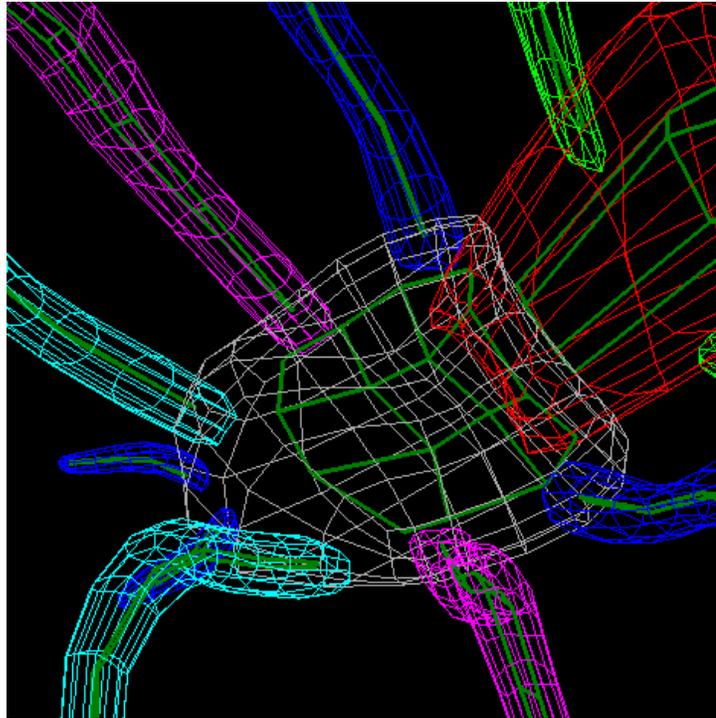


(a)

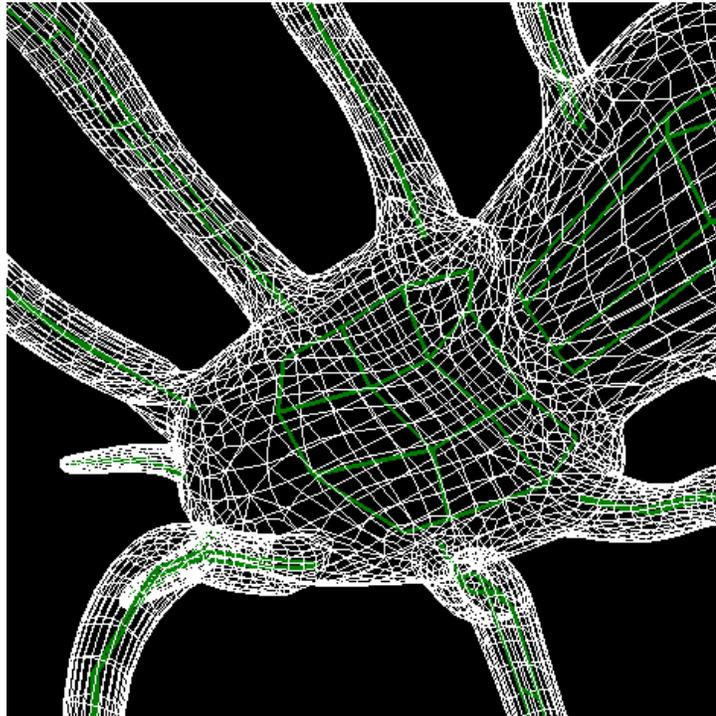


(b)

Figure 5.16: Subdivision boundaries for the spider m-rep; shown are (a) the individual-figure level-2 surfaces from Fig 5.15, solid shaded, and (b) the solid model rendered with blended figure-subfigure subdivision.

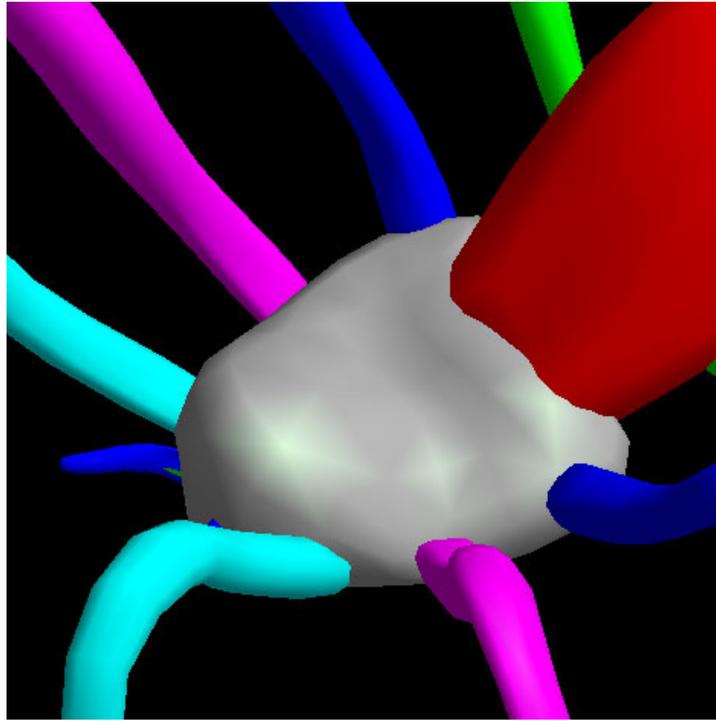


(a)

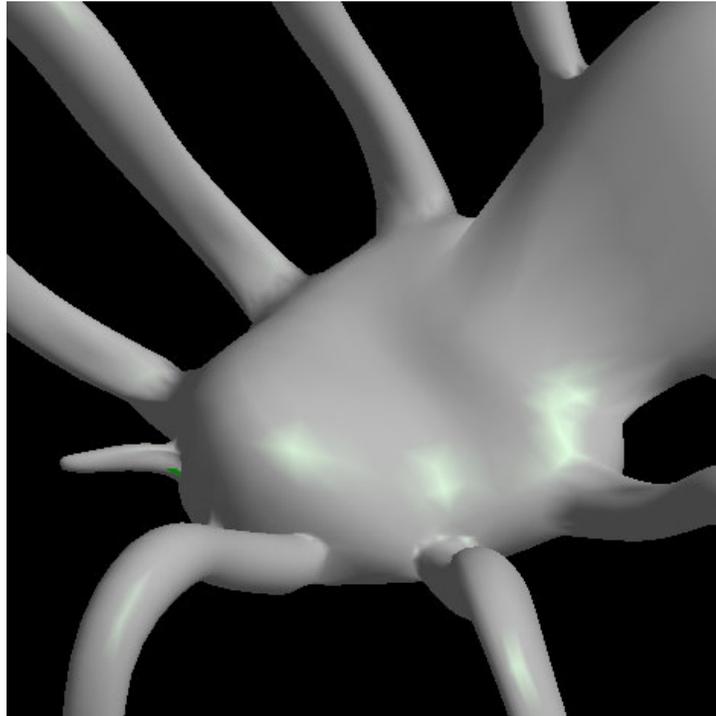


(b)

Figure 5.17: Close-up details of spider m-rep figure-subfigure blends; shown are (a) the unblended boundary meshes at level-1 and (b) the blended figure-subfigure meshes at level-2.



(a)



(b)

Figure 5.18: Close-up details of spider m-rep figure-subfigure blends; shown are (a) the unblended boundary surfaces at level-1 and (b) the blended figure-subfigure surfaces at level-2.

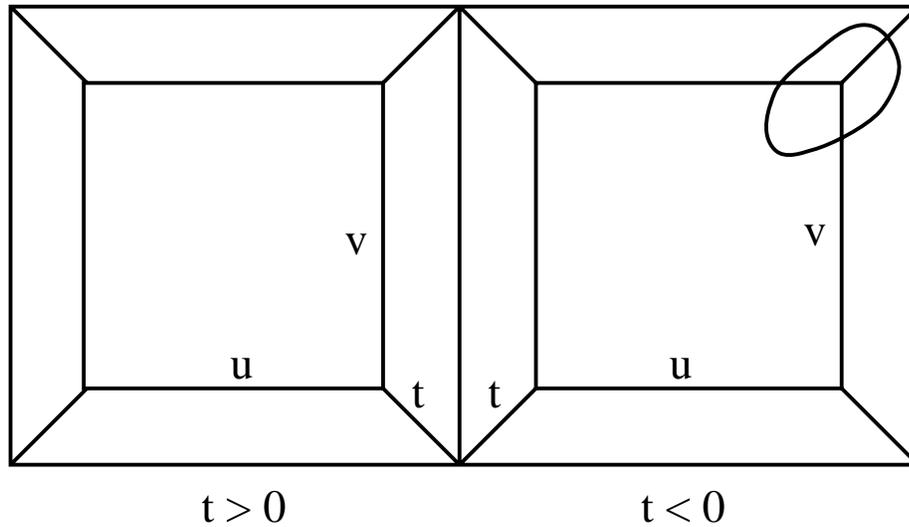


Figure 5.19: Positioning of subfigures across problematic crest regions.

there was no room on the cephalothorax to attach the rear legs without overlapping blend regions, so they were placed on the abdomen.<sup>2</sup> To address this problem, a reparameterization will again be required; the dilation of the intersection curve on the parent figure has to take into account local measures of tangency and curvature in the parametric domain.

One other feature which might be useful for general modeling tasks is to allow a subfigure to be joined obliquely to a parent figure, rather than restricting a subfigure join to be orthogonal to the subfigure’s medial axis. This would require subfigures to be attached without anchoring them at medial hinge atoms, simply by blending the surfaces of figure and subfigure. This would require a new means of producing subfigure deformations in response to parent-figure deformation but would allow m-reps to be treated similarly to convolution surfaces or other blobby modeling primitives.

### 5.3 Concluding remarks

This chapter demonstrated that the pieces are now in place for using m-reps in the manner for which they were conceived, as multiscale, multigure primitives for deformable solid modeling tasks in CAGD and for modeling and segmentation in 3D medical imaging applications. The next chapter will discuss two prototype m-rep

<sup>2</sup>Arachnologists should note that `big_spider.ply` also had faulty morphology—the apparent pedipalps in Fig. 5.13 are actually antennae.

modeling systems, *Rakshasa* (for CAGD) and *Pablo* (for medical image analysis). It will show how claims regarding m-rep effectiveness and advantages have been demonstrated using these systems, and it will discuss uses of m-reps by others that also support these claims.

# Chapter 6

## Discrete M-Rep Modeling Systems

Previous chapters have presented techniques for creating medially based multigure models. A number of prototype applications have been created to use discrete m-reps for object synthesis and deformation. This chapter examines two of these systems:

1. *Rakshasa*<sup>1</sup>, designed for object synthesis via m-reps for computer graphics and CAGD applications, providing a testbed for medial deformation, different medial mesh connectivities, boundary fitting and subdivision mesh displacement experiments;
2. *Pablo*<sup>2</sup>, designed by a MIDAG team for discrete m-rep modeling, segmentation and statistical analysis of objects from 3D medical image data.

The chapter will also discuss past and current research directions involving m-reps, including the work of Styner, of Fletcher and of Crouch, which further demonstrate the unique capabilities of m-reps in shape analysis and in physically based modeling. An examination of these modeling systems and research applications will support the claims made for discrete m-reps in Ch. 1.

### 6.1 Rakshasa: m-reps for computer graphics and CAGD

Rakshasa was a test-bed for early m-rep research, growing out of work on the earlier *Yaksha* system that supported experiments on slice-constrained medial primitives.

---

<sup>1</sup>from a shape-changing demon of Indian mythology

<sup>2</sup>named by Tom Fletcher, from Pablo Picasso. See epigram on Sec. 6.2.

Rakshasa was designed to support experiments on medial mesh topology and medially based deformation, on interpolation methods for boundary positions and for  $(u, v, t)$  and  $r$  values, and on boundary displacement and shading. It also served for early experiments in medial interpolation and subfigure attachment. It was the earliest of the m-rep modeling tools to make full use of the medial frame  $F$  information in the medial atoms  $\mathcal{M}_{\mathbf{p}} = \{\mathbf{p}, F, r, \theta\}$ , and it contributed large amounts of code to SCAMP and to Pablo’s Seurat library (see Sec. 6.2).

Rakshasa provided proof-of-concept for a large number of m-rep features and functions:

- the first medial atoms representing  $F$  explicitly, using quaternion frames;
- data structures for nets of medial atoms of different topologies, including simple quadmeshes, trimeshes, linear chains, circularly connected chains, and toroidally connected quadmeshes;
- subfigures linked to medial atoms of parent figures;
- deformations of objects based on modification of medial atoms in their local coordinate system, with changes propagated to the rest of the mesh atoms;
- techniques for smooth boundary interpolation;
- experimental techniques for smooth medial interpolation;
- an interactive GUI, using OpenGL and FLTK, for creating, modifying and displaying medial atoms, medial meshes, and boundaries;
- display functions for rendering slices from  $3D$  data sets based on cutting planes defined by the medial surface;
- the first (non-interpolating) subdivision boundaries for m-rep boundary definition and rendering;
- image and displacement textures on m-reps, producing boundary perturbations within the medially defined, width-proportional tolerances.

The discussion below will explore a number of these functions and features with an emphasis both on understanding implementation decisions and tradeoffs in Rakshasa (and derived Pablo code) and on discussing issues relevant to designing other m-rep systems, providing fuel for the discussion in Ch. 7.

### 6.1.1 Medial atoms and medial mesh topologies

Rakshasa implemented a `class Diatom` that encapsulated the elements of the medial atom  $\mathcal{M}_{\mathbf{p}} = \{\mathbf{p}, F, r, \theta\}$ , using a 3-vector for  $\mathbf{p}$ , a quaternion for  $F$ , and scalars for  $r$  and  $\theta$ . `Diatoms` also included change-of-basis code to switch between coordinate systems, from an atom’s  $\{\vec{b}, \vec{b}^\perp, \vec{n}\}$ -at- $\mathbf{p}$  frame to the world frame or that of another medial atom. These atom-based coordinates proved useful for defining atom-based medial modifications for m-reps, as described in the next section.

Rakshasa implemented a `class Mfig` as a parent class for different single-figure mesh topologies, including quad-figures, linear chain-figures and loop-figures, tri-figures, and tube and torus-figures based on quadmeshes. `Mfig` included virtual functions for rendering medial atoms, meshes and boundaries, and it also stored an instancing transform for positioning, orienting and scaling the object from its modeling coordinates. Although only the quadmesh structures were used in the subdivision boundary experiments, the development of the other data structures showed there would be no technical impediments to producing m-reps with other medial topologies for use in a full m-rep modeling system.

`Mfigs` also provided an array of links to other `Mfigs` for subfigure attachment; a link was attached to a medial atom in the parent mesh, with directions, distances, and orientations for the subfigure given in local atom coordinates. There was at the time no system for using  $(u, v, t)$  coordinates to position subfigures at the `Mfig` boundary—Rakshasa used  $(u, v, t)$  solely for boundary texture coordinates. Rudimentary code was in place for performing similarity transforms on linked subfigures based on parent-figure link-atom coordinates; these methods were made obsolete by the switch to subfigure hinge-atom linkage at parent-figure boundaries in Pablo.

### 6.1.2 Model construction and medially based deformation

Rakshasa allowed a user to specify a figural type—e.g., a quad-figure—and the scale and dimensions of the medial mesh—how many rows  $u$  and columns  $v$  of medial atoms. It then created a stock m-rep of the desired type. Modeling functions were limited; single atoms could be selected and have their attributes—position, scale, orientation or object angle—modified using simple GUI controls, or an entire row or column could be selected and modified. In addition, modifications could be made in the local coordinate system of a selected atom and propagated in selected directions along the medial mesh, allowing medially based deformations as primitive operations,

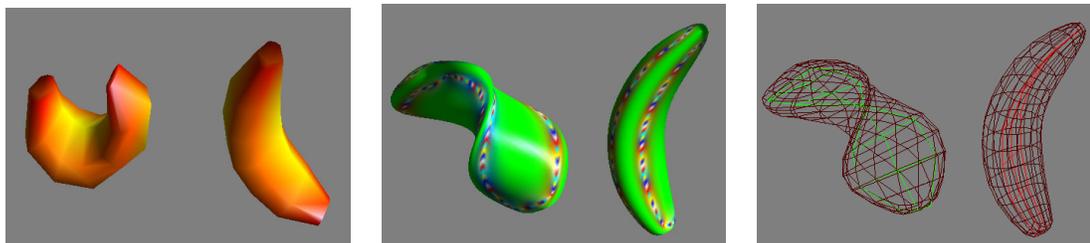


Figure 6.1: Curling and twisting operations on m-reps. As is the case for other skeletal modeling systems, m-reps can be manipulated by modifications applied to the medial skeleton. The use of medial-atom-based coordinates allows these deformations to be relative to the medial axis and medial frames.

curling or twisting or tapering the figure by corresponding operations on the medial mesh (see Fig. 6.1). Shape deformation by medial-axis modification is common to most skeletal modeling systems, but the use of medial-atom-based coordinates allows these modifications to be relative to the medial axis and its medial frames, rather than requiring external, object-independent coordinate systems or axes of rotation.

### 6.1.3 Medial and boundary interpolation

Rakshasa was used for experiments on both medial and boundary interpolation. Clearly, a medial interpolation of  $\mathcal{M}_{\mathbf{p}} = \{\mathbf{p}, F, r, \theta\}$  atoms over the medial surface based on an initial m-rep grid would allow the boundary to be interpolated directly from the medial surface in a Blum fashion. Considerable time was spent attempting constrained spline fits to the medial surface—work paralleled by Fletcher and later by Crouch—but the problem of correctly interpolating the medial radius  $r$  and frame  $F$  proved difficult without the sound theoretical bases developed later by Yushkevich, Damon, Fletcher and others. Interpolating a continuous medial surface raises its own issues, particularly in the need to maintain Blumness conditions on the surface; comparisons between discrete and continuous m-reps were made in Sec. 2.4.4.

My decision was instead to reject a medial-interpolation approach and directly interpolate boundaries based on the boundary involutes implied by the atoms of the coarse medial mesh. This approach would lead to the  $(u, v, t)$  coordinate system, establishing correspondence between medial and boundary coordinates parametrically rather than by Blum-based bitangency relationships. Early Rakshasa experiments used a spline-based approach—once again, paralleling work by Fletcher—but a subdivision-surface approach was chosen as a more general method, better suited to non-genus-0 and multifigure m-reps.

### 6.1.4 Catmull-Clark subdivision boundaries

Rakshasa produced the first application of subdivision surfaces to m-rep boundaries, using non-interpolating Catmull-Clark subdivision and restricting surfaces to single-figure quadmesh m-reps. The non-interpolating nature of the subdivision proved unacceptable, particularly for end-atoms, where shrinkage of the subdivision surface away from the boundary typically exceeded reasonable boundary tolerances. Likewise, the lack of normal interpolation at the involute positions meant that much of the medial atom information was simply thrown away. This was less a problem in Rakshasa than in the early implementation of Pablo, in which automatic perturbation of medial atoms during optimization-driven model-to-image matching required that changes in the  $\nabla r$  structure at a medial atom be reflected correctly on the boundary. As mentioned in Sec. 4.1.5, without interpolating involute-normals, the sampled medial axis reduces to a simpler chordal axis, throwing away the first-order information stored in each  $\mathcal{M}_p$ .

### 6.1.5 Image and displacement textures

Rakshasa implemented texture-mapping according to the interpolated  $(u, v, t)$  coordinates and used OpenGL’s texture mapping mechanism for display (see Fig. 6.2(a)). It also used  $\alpha$ -blended transparency to allow the medial mesh and its atoms to be visible along with the textured and Phong-illuminated boundary. It implemented displacement-mapping according to the same interpolated coordinates and the interpolated  $r$  and used a table-lookup into the displacement image to calculate the vertex displacements at any given subdivision level (see Fig. 6.2(b)).

As Rakshasa did not do interpolating subdivision, there were some inconsistencies in vertex displacements in going from coarse to fine subdivision levels. There was also no attempt to produce different resolution displacement textures for each level to address the problem of point-sampled displacement fields.

### 6.1.6 Drawbacks of Rakshasa

Aside from the limitations already mentioned, Rakshasa had a number of other deficiencies. It had a clumsy interface for atom selection and medial-based object perturbation—Rakshasa failed to demonstrate intuitive aspects of solid modeling using medial methods. Drawbacks in Rakshasa’s interface provided insights into better

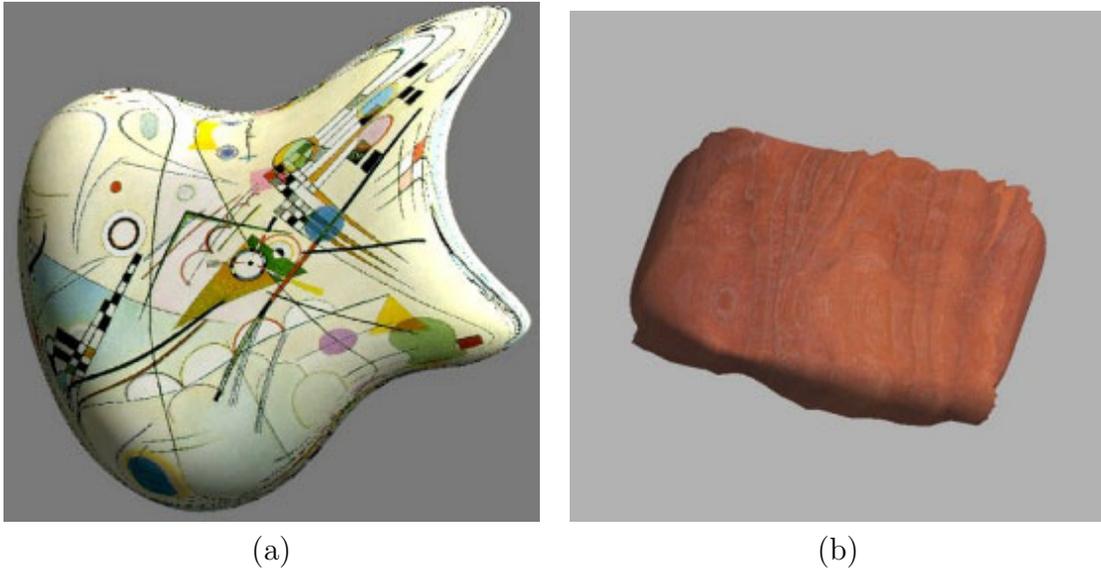


Figure 6.2: Image and displacement mapping in Rakshasa. Image (a) shows an image-texture on an m-rep subdivision boundary. Image (b) shows a woodgrain and displacement texture.

interaction mechanisms for Pablo and led to the concept of *representational transparency*, which provides the underlying theme for the discussion of future medial modeling systems in Ch. 7.1.1.

Rakshasa provided proof-of-concept for  $r$ -proportional displacement texturing of m-rep boundaries but never went beyond preliminary experiments. There were no interactive tools for editing boundary displacements, nor was there any attempt to use an actual displacement shader—e.g., by providing output using the Renderman RIB API. At present, the displacement mesh techniques fit better into the standard rendering APIs and provide explicit geometry for computational tasks; however, as procedural and displacement shading become more integrated with graphics hardware and software, the use of displacement textures for rendering will be an obvious way to reduce model-complexity for m-reps.

As mentioned above, Rakshasa did not fit subdivision surfaces to medial topologies other than the quadmesh—nor does Pablo. There are no major problems to face in doing so, although trimeshes would be better served by Loop than Catmull-Clark subdivision. New medial topologies would, however, require new medially based parameterizations. For many, these could be trivial modifications of the simple  $(u, v, t)$  coordinates of the quadmesh; for other topologies, they would be more problematic. These issues will be discussed in Sec. 7.2.5.

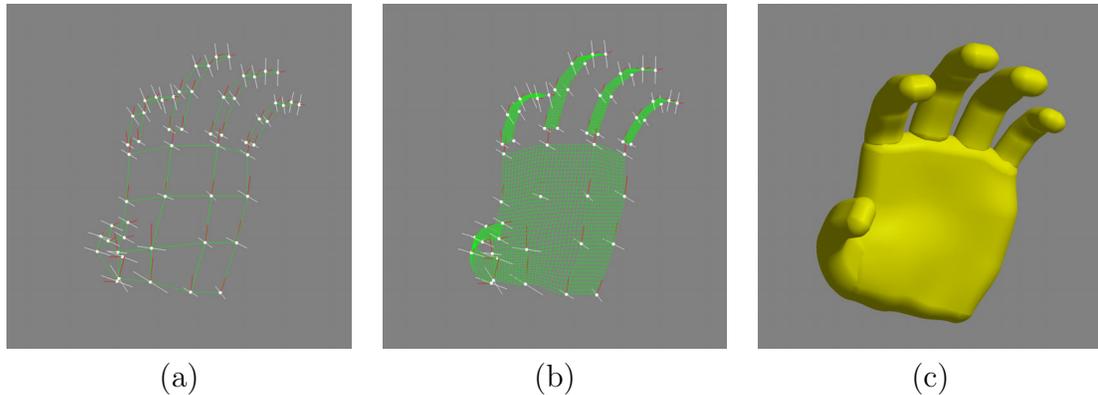


Figure 6.3: [Tom Fletcher] Prototype multifigure m-rep produced by SCAMP. This model shows a palm-figure with five finger-subfigures. Image (a) shows the coarse meshes of slice-constrained medial atoms. Image (b) shows a fine meshing of the medial surface using a spline-based positional fit. Image (c) shows the unblended figure and subfigure boundaries as generated by early spline-based code.

## 6.2 Seurat and Pablo: m-reps for modeling and segmentation in image analysis

“Computers are useless. They only give us answers.” —Pablo Picasso (from Tom Fletcher).

Pablo was created following prototype work on *SCAMP* by Fletcher, Thall, Fritsch and Fridman [58]. *SCAMP* used slice-constrained meshes of medial atoms for modeling purposes. The slice-constrained atom requirement allowed simplified extrusion-modeling and boundary interpolation but proved too limiting for general modeling purposes. *SCAMP* did produce some of the first multifigure models, however, as in Fig. 6.3, and created animations of them using medial deformation [148].

The Pablo application was created as a testbed and prototype application for m-rep-based modeling, segmentation and statistical analysis of 3D medical images from CT or MRI. It grew out of initial work on *DSL*<sup>3</sup> modeling under Daniel Fritsch. Pablo is the flagship platform for m-rep-based modeling and segmentation work at MIDAG, and the Seurat library, my main contribution to it, encapsulates all functionality for boundary interpolation and rendering, mappings between  $(u, v, t)$  or  $(u, v, t, \tau)$  and  $(x, y, z)$ , and figure-subfigure attachment by remeshing. This section will begin with a brief discussion of Pablo and its capabilities and then discuss the contribution of

<sup>3</sup>*deformable shape loci*, an old name for an early version of m-reps

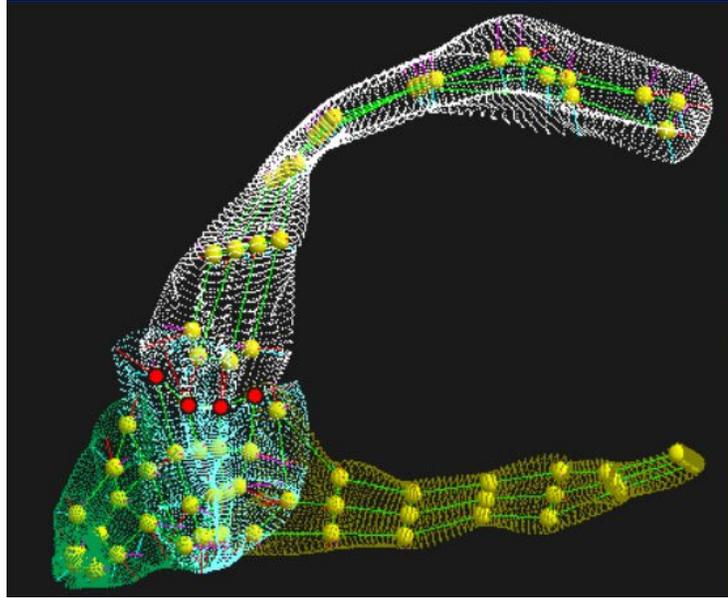


Figure 6.4: Multifigure brain-ventricle m-rep modeled in Pablo. This model of the lateral ventricle was created by Gregg Tracton, Stephen Pizer and Guido Gerig.

the Seurat library to these capabilities.

### 6.2.1 Pablo functionality

Pablo is a research tool and code-base for applying m-reps to medical image segmentation, particularly directed at tasks in 3D computed tomography and MRI data analysis. Pablo allows the hand-building of multifigure m-rep models, letting the user create and manipulate m-reps and position them relative to an orthogonal-slice-based view of a 3D dataset. It also allows an atom-based view of the dataset, rendering the  $\{\vec{b}, \vec{b}^\perp\}$  of the atoms in-plane with a slice through the data. User interaction for manual shape-construction is via a GUI-based, click-and-drag interface, allowing atoms to be selected and modified by translation of  $\mathbf{p}$ , rotation of  $F$  (by virtual trackball), rescaling of  $r$  or change in  $\theta$ . If a group of atoms is selected (using mouse-driven, rubber-banded box selection), rotations and rescalings are made about a common averaged center. Pablo's m-reps are bounded by IIS-surfaces, with user-assignable tolerances on boundary normal interpolation. Fig. 6.4 shows a multifigure m-rep built using Pablo; Fig. 6.5 shows a medial mesh with coarse involute mesh, along with interpolating subdivision boundaries at different subdivision levels.

Along with the hand-building of m-rep models, Pablo implements a technique for medical image segmentation using these m-reps under a multiscale optimization

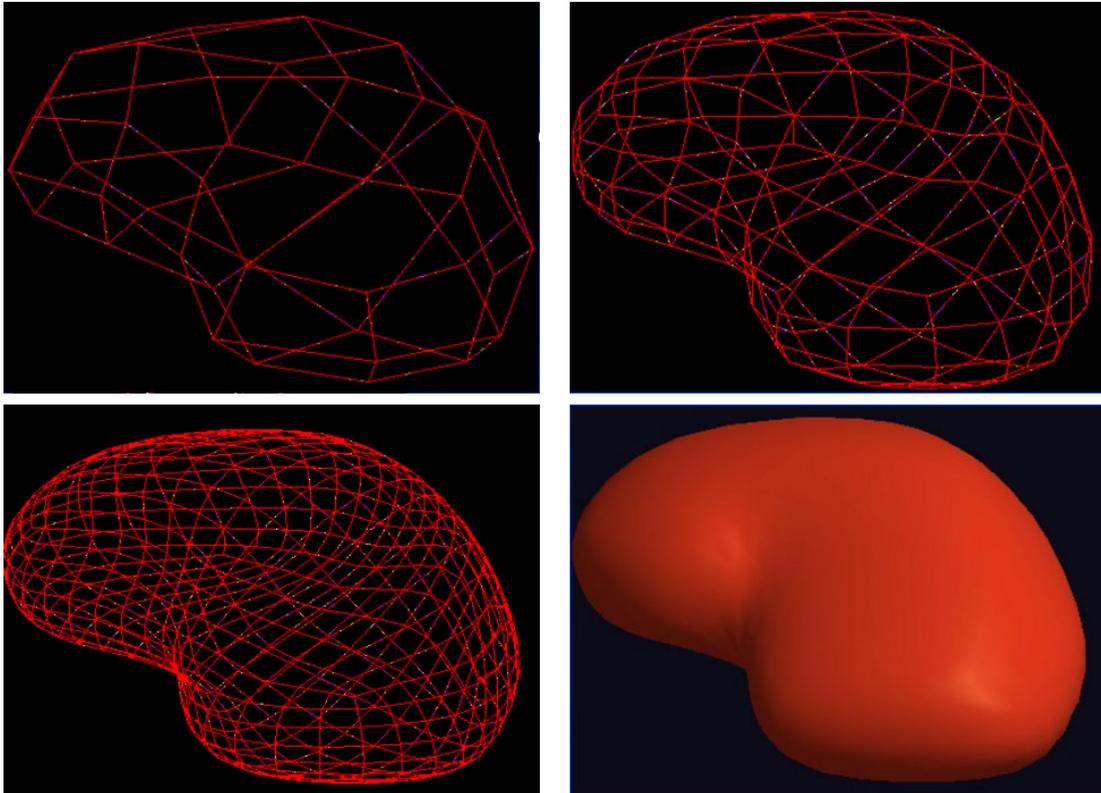
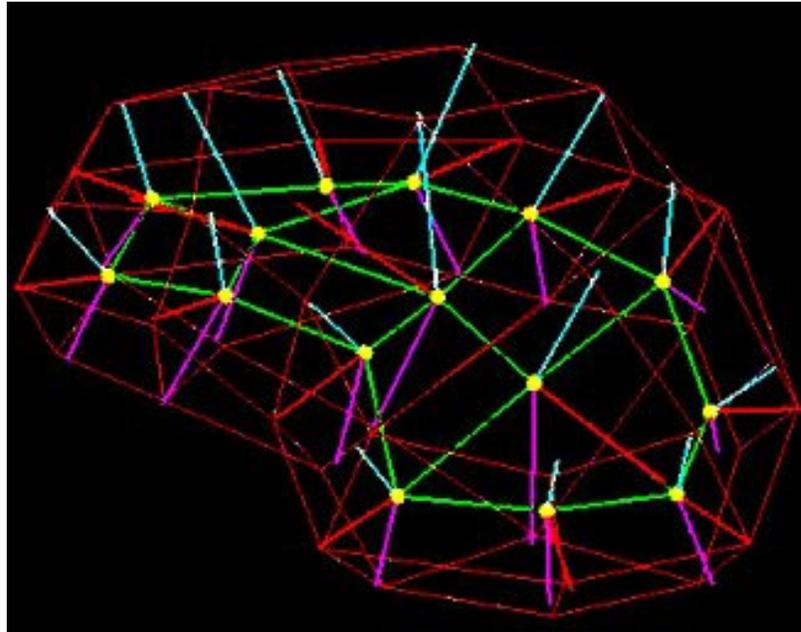


Figure 6.5: Pablo kidney m-rep with subdivision boundaries. At the top is a single-figure kidney model with medial mesh, atoms and coarse boundary mesh; below it are interpolating subdivision surfaces of increasing subdivision level along with a smooth-shaded final model.

paradigm. Based on an initial manual placement, an m-rep is deformed in multiple optimization stages:

1. by similarity transform of the entire m-rep;
2. by similarity transform and figural elongation of the main figure;
3. by similarity transform, hinging and elongation of subfigures in main-figure coordinates;
4. by medial atom perturbation within figures and subfigures;
5. by boundary vertex displacement along medially implied normals.

The details of this are discussed in Pizer et al. [121]. The optimization is based on both image-intensity match and statistical priors based on geometric typicality measures using a Bayesian deformable templates methodology. This allows prior knowledge of geometry and expected shape variability to guide the deformations. Fig. 6.6 shows two views of a kidney m-rep in a 3D dataset rendered with orthogonal cut planes.

The implementation of m-reps in Pablo was based primarily on the work of Tom Fletcher and me. Fletcher wrote simplified versions of the medial atoms and quadmeshes for use in the m-rep manipulation and analysis routines of Pablo; my own m-reps, derived from the Rakshasa code, were used in the boundary generation and displacement subdivision code and in the interpolation routines. Pablo demonstrated a number of new features and mechanisms for medial modeling:

- medial end atoms with an explicit  $\eta$  term for representing elongation in crest regions;
- the use of medially based coordinates for parametric correspondence between boundary  $(u, v, t)$  and medial  $(u, v)$  locations;
- the use of subfigure hinge atoms to link subfigures to boundary  $(u, v, t)$  positions on parent-figures, with the ability to modify subfigures based on the parent-figure coordinates;
- the use of IIS-surfaces for boundary fitting, with mesh vertex displacements representing boundary perturbation;
- an implicit-surface technique for smoothly joining the subdivision boundaries of figure and subfigure;

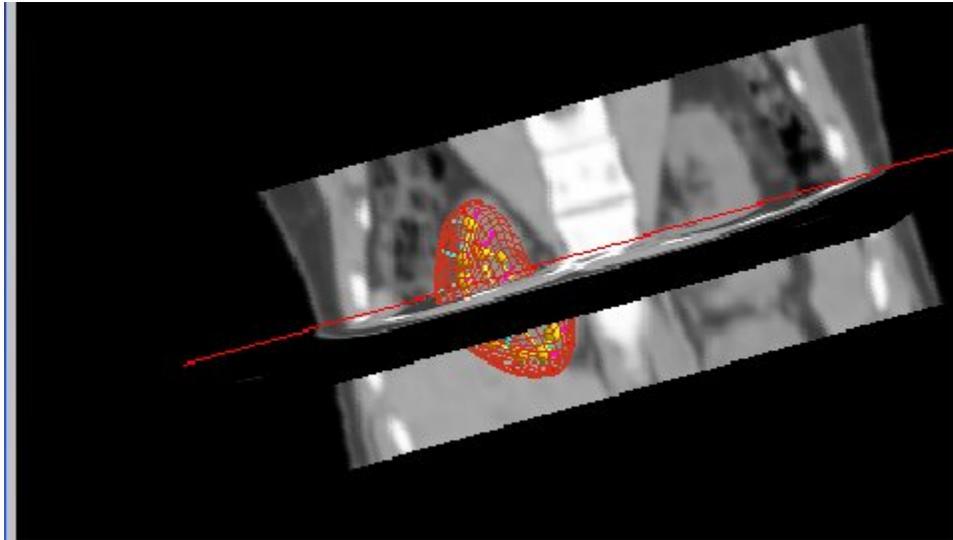
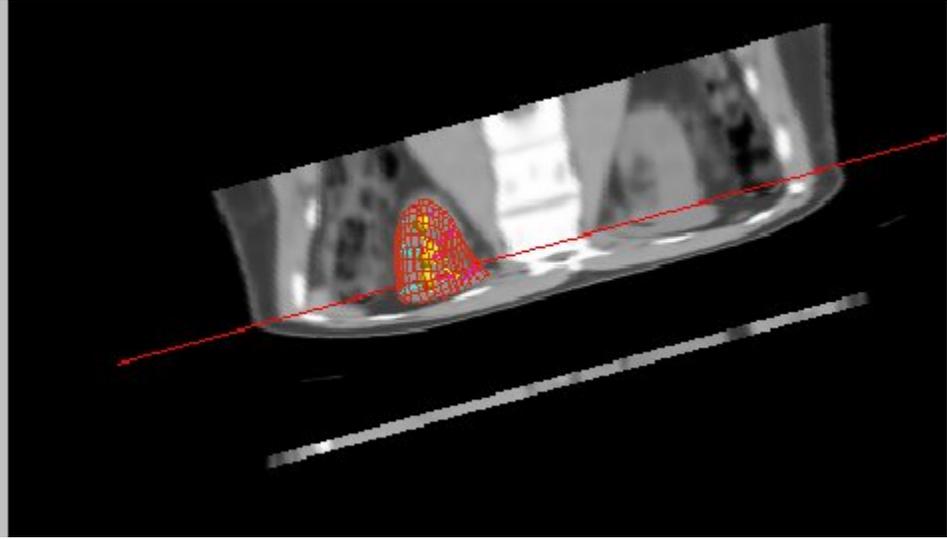


Figure 6.6: Two views of a Pablo kidney m-rep in image data with cut-planes.

- a subdivision surface remeshing technique for smoothly joining subdivision boundaries of figure and subfigure, allowing a filleted join region with specifiable curvature properties;
- CSG-style addition and subtraction operations using figure-subfigure hierarchies;

Pablo can thus be seen as a fairly complete implementation of the m-rep capabilities discussed in Chs. 3–5. The interface lacked the user-driven medially based deformations found in Rakshasa, but the use of mouse-interaction for m-rep manipulation, as implemented by Fletcher, made simple modeling tasks more straightforward. Nonetheless, the lack of intuitive solid modeling operations is still evident; Sec. 7.1 will discuss better approaches to deformable solid modeling using m-reps.

### 6.2.2 Seurat functionality

In order to bring capabilities from Rakshasa into the Pablo project, a library of supporting classes was derived from it, stripping out rendering-specific and experimental deformation code to leave base classes for medial atoms, m-rep quadmeshes, and boundary interpolation and parameterization schemes. This class library served as the foundation for the Seurat library, which became an key component of the Pablo project.

Seurat was devoted to medial-to-boundary representation issues; this discussion will touch on the most important of its operations:

- IIS-surface fitting to m-reps;
- mapping between  $(u, v, t)$  and  $(x, y, z)$  on the boundary and between  $(x, y, z)$  and  $(u, v, t, \tau)$  in space;
- legality checking to prevent locally self-intersecting boundaries;
- proximity/nearpoint testing for IIS-surfaces;
- figure-subfigure joining by smoothly remeshing IIS-surfaces.

Each of these implemented techniques merits a brief discussion.

## IIS-surface library

The subdivision code inherited from Rakshasa was non-interpolating and lacked generality; it could only fit subdivision surfaces to genus-0 boundaries with all-quad tilings. Once this was ported to Seurat, there began a long process of modification and rewriting, culminating in a new, more general library of IIS-surface routines allowing interpolation of m-rep boundary positions by the modified Catmull-Clark software, with prototype code in place for interpolating boundary normals as well. To utilize the full power of Catmull-Clark methods at irregular vertices, the techniques described in Sec. 4.1.3 were implemented, allowing IIS-surfaces to be created for irregular meshes. The current subdivision code has the following characteristics:

1. it is class-structured C++ code, integrated with Seurat and Pablo but easily made stand-alone;
2. it accepts unordered mesh data input as vertex/polygon lists and allows polygons with three or more vertices;
3. it determines connectivity information and constructs data structures for subdivision;
4. it performs interpolating or non-interpolating subdivision with or without normal interpolation to within user-specified positional and rotational tolerances.

Seurat also allows displacements at each vertex, though the mesh code in Seurat does not currently recalculate boundary normals based on perturbed vertex positions.

## Medial coordinates and mappings between medial and Euclidean coordinates

Seurat was a testbed for single-figure  $(u, v, t)$  and  $r$  interpolations, with the eventual solution using a bisection algorithm during the Catmull-Clark splitting phase. Various spline-based approaches were also attempted, as well as using the Catmull-Clark splitting itself; these methods all suffered from coordinate-creep near crest regions. Along with  $(u, v, t)$  interpolation, functions were created to compute  $(u, v, t) \mapsto \mathbb{R}^3$ , identifying boundary points at a given subdivision level corresponding to a given  $(u, v, t)$  value, as well as a generalized inverse  $\mathbb{R}^3 \mapsto (u, v, t, \tau)$ , giving the medially based coordinates for a point in  $\mathbb{R}^3$ . The latter was well-defined only internal to and

within a neighborhood of the m-rep boundary; the mapping was elsewhere problematic. A true medial coordinate system requires a non-linear mapping to the surrounding space, creating a distance-at-scale function in  $\mathbb{R}^3 \times \sigma$ ; this can be conceptualized as analogous to the geodesic distances in a curved space-time found in general relativity. The linearized  $(x, y, z)$ -to- $(u, v, t, \tau)$  functions are adequate for their intended use as localized,  $r$ -proportional distance functions over the subdivision boundary for image-segmentation tasks and for implicit-surface blending.

As an implementation detail, surface positions over the tiles were computed based on bisecting each quad at the given subdivision level to form a pair of triangles with the common diagonal between the  $(u, v)$  and  $(u + h, v + h)$  vertices in the quad. This splitting is used for the interpolation of  $(x, y, z)$ ,  $\vec{n}$ ,  $r$ , and  $(u, v, t)$  across tiles at a given level. If a quad were too non-planar—generally, only at the coarsest subdivision level—this splitting could create problems for the nearpoint code. The solution in that case was generally to subdivide once to better define the surface. An alternate method to this scheme is discussed below in Sec. 6.2.3.

### Legality checks on meshing during modeling operations

Seurat implemented a number of legality checks on m-rep medial meshes that were applied during automatic model-to-image matching operations in Pablo. Problems arose when medial atoms were perturbed in ways that introduced self-intersections or unacceptably high curvatures at the interpolating subdivision boundaries. Seurat included four tests:

1. concavity—for each of the 4 possible quads containing the atom at  $(u, v)$ , a concavity check is done by projecting the three other atoms onto the plane of the test atom and checking the resulting planar quad for concavity;
2.  $\Delta r$ -to- $\Delta s$ —for each of the 8 possible neighbors of a test atom, an error is returned if  $\Delta r$  is greater in magnitude than  $\Delta s$ ;
3. non-planarity—an error is returned if the test atom’s  $\{\vec{b}, \vec{b}^\perp\}$  plane is too skewed with respect to the neighboring atom positions;
4. crossed-involutes—an error is returned if there is overlap of the  $\vec{v}_1$ ,  $\vec{v}_2$  or  $\eta\vec{b}$  involute vectors of an end-atom and its neighbors along the crest.

With regard to the concavity test, the concavity of a projected quad may not give a good indication of how badly askew the actual quad is. If one assumes nearly

planar quads, however, this test is generally good. It is acceptable that these tests be conservative, returning rejection for marginal cases.

Other methods for legality checking used boundary curvature penalties to prevent the morphogenesis of corners and creases during deformation. These methods used tests for *Blumness*, based on the work by Fletcher, Yushkevich and Damon discussed in Sec. 3.3.1. Legality checks are a requirement for automatic segmentation; for future applications, in CAGD tools for m-rep design, legality checks may be used to restrict acceptable deformations or signal the user when a model has been perturbed beyond tolerances (see Sec. 7.1.1).

### **Proximity/nearpoint testing in Seurat**

Seurat implemented the proximity tests discussed in Sec. 4.2, initially for use in an implicit function for figure-subfigure blending. The tests have since proven generally useful for finding figure-subfigure intersections (for remeshed blending) as well as for the computation of Blum medial correspondence on the subdivision-bounded solids, as per Sec. 4.3. These tests currently assume an unperturbed boundary; for modeling purposes, it may be adequate to simply use the correspondence between perturbed and unperturbed  $(u, v, t)$  positions; for physically based modeling, collision-detection and other applications, the theory and implementation of multiscale proximity tests remain both enticing and problematic.

### **Multifigure modeling by remeshing**

Current development in Seurat centers on multifigure construction using the remeshing technique created by Qiong Han and me, with rewritten IIS-surface libraries being fully implemented and tested in the current Pablo code-base.

On the model-building side, remeshed subfigures allow better m-rep creation of multifigure models, with smoothly filleted joins for additive and subtractive subfigures, and allow creation of m-reps with non-genus-0 topologies. On the segmentation side, Pablo can now employ the full power of hierarchical and graph-based modeling for object deformation and registration, performing matching over the entire m-rep boundary, including remeshed blend regions, and using parent-figure-based coordinates to perturb subfigures in automatic segmentation. Subfigure optimization can now proceed by running several iterations over the subfigure without the blend region, followed by further iterations including it. Perturbations of a subfigure based

on propagation of changes to its hinge-atoms by parent-figure perturbation will best be handled by the Lie algebra techniques currently under development.

### 6.2.3 Drawbacks in Pablo and Seurat

There remain a number of limitations in the m-rep implementation in the current Pablo and Seurat libraries.

1. The interpolation of boundary normals at involute positions remains unsatisfactory in requiring care on the part of the user to prevent rippling effects. A relaxing of the tolerance on the acceptable  $\theta$ -deviation at boundary involutes will reduce ripples, but a better solution will likely come either from curvature-constraints methods such as Halstead's or from other methods for fairing multiresolution meshes. Rippling presents problems both for segmentation, where boundary normals are crucial, and for projected CAGD applications.
2. As previously discussed, quad tiles are bisected for bilinear interpolation of  $\vec{n}$ ,  $r$ , and  $(u, v, t)$  across tiles and for nearpoint computation. For near-planar quads, the choice of diagonal for the bisection makes little difference; still, it adds an unnecessary element of arbitrariness to the process. Direct use of explicitly triangulated boundaries would eliminate this problem; it could be accomplished by a shift to Loop subdivision, augmented by the same IIS-surface interpolation used presently for Catmull-Clark subdivision. Another alternative would be to fit a higher-order interpolation patch to the boundary quads.
3. Vertex displacements have been implemented in Pablo and are used during model-to-image fitting; they are not yet integrated into the modeling program, however. Pablo needs tools to allow a user to manually create boundary perturbations for models; currently built models often depend on accidents of the surface interpolation, ripples and all, to produce boundary characteristics that should rather be fine-scale boundary detail. Vertex displacement are also not considered by the current nearpoint/proximity tests.
4. Model-building in Pablo is still awkward—the click-and-drag method for selecting and manipulating multiple atoms is better than in Rakshasa but still not an intuitive shape design method, and multifigure modeling in Pablo lacks dedicated design tools for positioning and manipulating additive and subtractive subfigures. Better interface paradigms are needed, perhaps drawing from

sketch-based interfaces or force-feedback driven virtual sculpting tools. Pablo also provides little support for m-rep polymorphism, using only quadmesh slabs as figural primitives.

5. Pablo’s legality checking is still primitive and might benefit from the use of Blunness measures, as described above, or from the use of Damon’s medial shape operators (see Damon [39]) to provide a continuous measure of “near-foldedness” better suited to optimization routines than binary or heuristic tests.
6. Pablo lacks the ability to produce a multiresolution resampling of the medial mesh. The ability to adjust the medial sampling frequency and express a fine mesh as residues of a coarser mesh would provide a coarse-to-fine description of medial shape. This is not required for shape-description purposes but offers improved efficiency both for statistical analysis and for segmentation. For m-rep meshes, mesh-neighbor relationships can be used to reduce  $\mathcal{O}(n^2)$  every-atom-to-every-other-atom computations to  $\mathcal{O}(n)$  computations over local mesh neighborhoods. Using a coarse-to-fine mesh structure allows these local neighborhoods to provide their information at multiple levels of detail.

None of these drawbacks have prevented m-reps from being useful multiscale modeling primitives, and Ch. 7 will discuss methods for eliminating them. A last drawback, which has a negative impact on m-rep statistics, is the fragility of the quaternion-based m-rep frame when the object-angle  $\theta$  approaches  $\frac{\pi}{2}$  for near-parallel object boundaries. Because slab-like regions are commonplace, statistics on medial frames are often disrupted by their discontinuous change in these regions, even while the important shape characteristics show no such discontinuities. Current research by Fletcher at MIDAG extends earlier work (see Sec. 6.3.4) to employ statistics directly on the *sail vectors* connecting the medial atom center-points  $\mathbf{p}$  to their boundary involutes.

### 6.3 M-rep research applications

M-reps have become a major part of modeling, segmentation and registration systems for medical image analysis within MIDAG and have been widening their exposure in the image analysis community at large. At this point, it is appropriate to discuss several research directions that have employed m-reps to achieve results lending credence to claims for m-rep power and utility. This section will first cite the current Pablo

work within MIDAG and then examine three other research directions involving m-reps: the 3D shape analysis studies of Styner and Gerig, the FEM-based deformation techniques of Crouch, and the use of Lie groups to analyze m-rep deformation leading to the *principal geodesic analysis (PGA)* of Fletcher. Each of these research directions illustrates different aspects and strengths of m-rep-based object primitives.

### 6.3.1 Current Pablo research

Pablo, with the functionality as described above, has been a testbed for 3D model-based medical-image segmentation and shape-analysis in MIDAG. The segmentation research uses m-reps to perform automatic segmentation based on an initial manual placement of a model in a data set; from this initial position, Pablo then performs a series of optimizations as discussed previously in Sec. 6.2.1. The optimizations take place from large scale to small:

- at the multi-object, object and figural scales, with similarity transforms and variations over principal medial warp modes;
- at medial atom scale, using atom-based medial modification;
- at fine boundary scale using boundary perturbation.

All of these use both image-match terms and geometric priors. This multiscale segmentation is made possible by m-reps' separation of objects into figural hierarchies, by their medial nature allowing bending, bulging and elongation to be expressed globally over the implied medial sheet and locally at individual medial atoms, and by their local, fine-scale variability afforded by the subdivision boundaries.

M-rep-based segmentation has been validated in studies of kidney CT data by Rao [126]. Based on a training set of 60 hand-segmented CT images, mean left and right kidney models were created and principal geodesic modes of variation were computed for them. Intrinsic  $(u, v, t, \tau)$  coordinates were used to parameterize intensity-profile templates on the boundary. These templates, based on work of Joshua Stough at UNC, became an option to a first-derivative-of-Gaussian (*DOG*) in  $\tau$  profile with local templates derived from three different profiles: a *DOG* from dark-to-light, a *DOG* from light-to-dark, or a shifted dark Gaussian. The template value at a given  $(u, v, t)$  was the profile seen as most prevalent at that location across the training set. Using medial warps and the new boundary templates, Rao was able to quantify improved segmentation over Gaussian-template-based segmentations, and the

segmentations now required no case-by-case parameter adjustments. In most of the cases (22 of 24), the resulting segmentations would require no further editing for use in clinical applications. Currently, research is underway to test both the efficiency and accuracy of m-rep-based segmentation versus segmentation by more mainstream active shape models.

Detailed discussions of the state-of-the-art in multiscale deformable model based segmentation using m-reps can be found in papers by Joshi et al. [87, 86] and by Pizer et al. [123, 121].<sup>4</sup>

Much of m-rep shape-analysis work has focused on statistical discrimination studies of hippocampus and caudate nucleus structural changes in the brains of schizophrenic patients. This includes the work of Styner, Gerig et al. [145], of Gerig, Muller et al. [68] and of Vetsa et al. [153]. The work of Styner et al. distinguished healthy from non-healthy brain-structures; the work of Gerig, Muller et al. was able to demonstrate correlations based on types and duration of treatment in unhealthy patients. The work of Vetsa, similar to Styner and Gerig, presented statistical analysis of caudate nucleus structure in schizophrenia patients, using m-reps to allow global and local growth versus deformation to be tested separately and without the setting of case-by-case parameters. These were exploratory statistical analyses, requiring further, hypothesis-driven clinical studies to establish full confidence. Nonetheless, the results are promising.

In both segmentation and shape-analysis, m-reps demonstrate their advantages in defining shape deformations based on local shape descriptions of bending/twisting and bulging/compressing—as opposed to non-local techniques such as spherical harmonics or local (e.g., active-shape) techniques measuring only translational differences. M-reps also provide their medially based coordinate systems within which to do geometry-to-intensity matching or, using variation in  $\tau$  at a given  $(u_0, v_0, t_0, \tau)$ , to allow profiles to be established to create templates for expected boundary behavior. With smoothly meshed multigure m-reps presently being integrated into Pablo, with displacement meshed subdivision boundaries becoming more sophisticated and with the new statistical techniques being developed for m-rep analysis, Pablo is at the point now of realizing the full potential of discrete m-reps as robust shape descriptors for 3D model-based image analysis.

Lastly, another Pablo-based m-rep research project is the heart-atlas modeling work of Pilgram and Schubert at Innsbruck [119]. This work uses m-reps for their

---

<sup>4</sup>These are available online at <http://midag.cs.unc.edu/pubs/papers>.

multifigural nature and their intrinsic parameterization of multiscale shape variability. This is work in progress and will in the future incorporate finite-element meshing of objects and interstitial spaces using techniques for m-reps described below in Sec. 6.3.3.

### 6.3.2 Styner: m-reps for statistical shape analysis

Styner was a key player in the MIDAG research on m-rep-based statistical analysis of brain structures and the associated statistical discrimination studies described above. In his studies, he compared spherical harmonic (*SPHARM*) analyses with m-rep driven ones (atom-by-atom, with no global analysis) and was able to show the same level of discrimination with each. The advantage he showed to m-reps was their additional ability to localize deviations and to isolate bending and elongation changes.

Styner also developed a method for statistical training of m-rep models from a population of objects described by SPHARM surface descriptions [146, 144]. Styner first performed principal components analysis (*PCA*) on his training set of SPHARM models. Based on these statistics, the shape space was sampled to produce a representative object set, and then Voronoi skeletonization was used to compute medial branching topologies for each member of the representative set. From these individual topologies, after pruning, a common branching topology was derived by mapping them into a common spatial frame. A discrete m-rep model was then created that produces a best fit to this topology. This model could be automatically fitted to image-data using a modified version of Pablo.

Styner’s statistical work demonstrated the benefit of m-reps’ separation of *growth* and *bending* characteristics, and his training algorithm illustrated the simplification afforded by moving from a continuous Voronoi skeleton to a discrete m-rep, eliminating the instabilities in the medial branching topology by replacing the Voronoi axis with an optimal m-rep.

### 6.3.3 Crouch: finite element modeling using m-reps

Crouch [35, 33] used multifigure m-reps to model human tissue deformation. M-rep capabilities were used in three separate ways. First, FEM requires a 3D mesh structure for objects; the medial structure of m-reps was used to develop automatic techniques to produce hexahedral meshes for single and multifigure m-reps. For single-figure

m-reps, the technique used  $(u, v, t, \tau)$  coordinates to create a regular grid structure. For multifigure models, the technique needed not simply a boundary remeshing to attach a subfigure—as required in Pablo—but a spatial transition gridding between parent and child figures to give a regular and unbroken hexahedral grid structure for the entire m-rep. As implemented, these methods only work for protrusions and not indentation figures. Second,  $(u, v, t)$  coordinates were used for geometric correspondence between initial and deformed models, with corrections to give estimates of physical correspondence; these correspondences were used as boundary conditions for the FEM deformation. Third, a multiscale approach based on  $(u, v, t, \tau)$  was taken to the solution of the FEM equations. A coarse gridding of the objects produced an initial solution, which was then used as input for an FEM solution on a finer, m-rep subdivision-based remeshed grid. This was carried to a third subdivision level, though the improvement from second to third was insignificant.

These methods were applied to the driving problem of modeling prostate deformation for the implantation of radioactive seeds (*brachytherapy*) for cancer treatment. The use of medial skeleta for FEM methods was explored earlier by Storti et al. [142], as mentioned in Sec. 2.1.2; the use of m-reps combines medial FEM methods with multiscale shape methodology, bringing the advantages of the latter to tasks such as automatic registration and segmentation in physically based deformation of tissues during treatment planning and execution.

### 6.3.4 Fletcher: Lie algebras and principal geodesic analysis on m-reps

Fletcher et al. [55, 56] sought a way to do correct statistics on m-reps, a process hindered by the fact that the medial atoms  $\mathcal{M}_{\mathbf{p}} = \{\mathbf{p}, F, r, \theta\}$  do not form a vector space and thus create problems for methods such as PCA. Instead, the authors realized that medial atoms  $\mathcal{M}_{\mathbf{p}} = \{\mathbf{p}, F, r, \theta\}$  form Lie groups and could thus be analyzed using Lie algebra methods to determine Gaussian probability distributions over Lie groups. This allowed Gaussian distributions to be derived for a whole atom  $\mathcal{M}_{\mathbf{p}}$  and for meshes of atoms by closure of Lie groups under direct products.

Based on these computable distributions, Fletcher et al. developed a method of *principal geodesic analysis (PGA)*. Just as PCA finds eigenvectors in a linear  $nD$  feature space, PGA finds principal geodesic directions in the  $nD$  Lie group representing variations over populations of m-reps. Using PGA, a population of m-reps can

analyzed and principal modes of variation identified. The beauty of PGA is that it can be applied to myriad other objects as well as m-reps; fundamentally, it allows statistical characteristics of orientation to be incorporated into the same analysis as positions. The fact that medial atoms describe local position, scale and orientation and that m-rep medial meshes contain first-order shape descriptive information means that these shape properties can be explicitly captured by PGA methods.

The ability to do general statistics on Lie groups goes beyond simply allowing the finding of means and principal geodesics; the Lie group nature of m-reps provides the means for applying more general statistical techniques as well, such as allowing statistical discrimination studies to fit sampled data distributions.

Lie group methods have been applied to m-reps generated by automatic segmentations in Pablo, using PGA to analyze modes of variation in kidney and hippocampus populations. In turn, these PG components are used to provide modes of deformation with optimized coefficients; they can also be used to provide a log prior for the segmentation objective function. The ability to create meaningful probability distributions over populations of m-reps is essential for shape analysis tasks and desirable for computer graphics applications, as well, where they allow automatic generation of object instances with defined statistical variability. This is a subject of current exploration in the heart-atlas work of the Pilgram and Schubert [119].

### 6.3.5 Application of Fletcher’s methods for m-rep animation

As a proof-of-concept for the usefulness of Lie algebra methods on m-reps for computer graphics and animation, Qiong Han animated the spider m-rep we created and described in Sec. 5.2.4. The m-rep was manually altered in Pablo to create three different poses (see Fig. 6.7), and then interpolation on the Lie group manifold was used to automatically morph the m-rep between these three poses—in animator’s terminology, to automatically *in-between* the *keyframes* of the animation. Each morphed m-rep was modified by a random noise function (Perlin noise) added to the position and orientation of the end-atoms at the tip of the spider’s legs. The sequence of m-reps was then rendering with a shifting and rotating viewpoint to make the moving spider appear to be walking. The resulting animation was surprisingly life-like considering the lack of physically-correct dynamics or kinematics. Han speculates that applying the noise function to the atom positions, orientations and radii in the rest

of the model might further enhance the visual realism.<sup>5</sup> The current work demonstrates that automatic medial deformation of m-reps allows figurally-based, skeletal deformations to simulate physically realistic changes and movement in a solid model.

## 6.4 Concluding remarks on modeling systems

This chapter has discussed past and current implementations of m-reps that demonstrate the prototype modeling and deformation tools and illustrate the utility of m-reps as multiscale shape descriptors. It discussed several applications of m-reps that also illustrate their usefulness and the power gained by providing a medial shape descriptor separating coarse-scale object shape variability from fine-scale boundary structure.

The next chapter will discuss the tools and modeling paradigms appropriate for m-rep modeling for CAGD and shape analysis applications. It will explore improvements in m-rep technology and directions for future research and applications.

---

<sup>5</sup>Information on the use and usefulness of Perlin noise to enhance realism in simulations can be found at <http://mrl.nyu.edu/~perlin>.

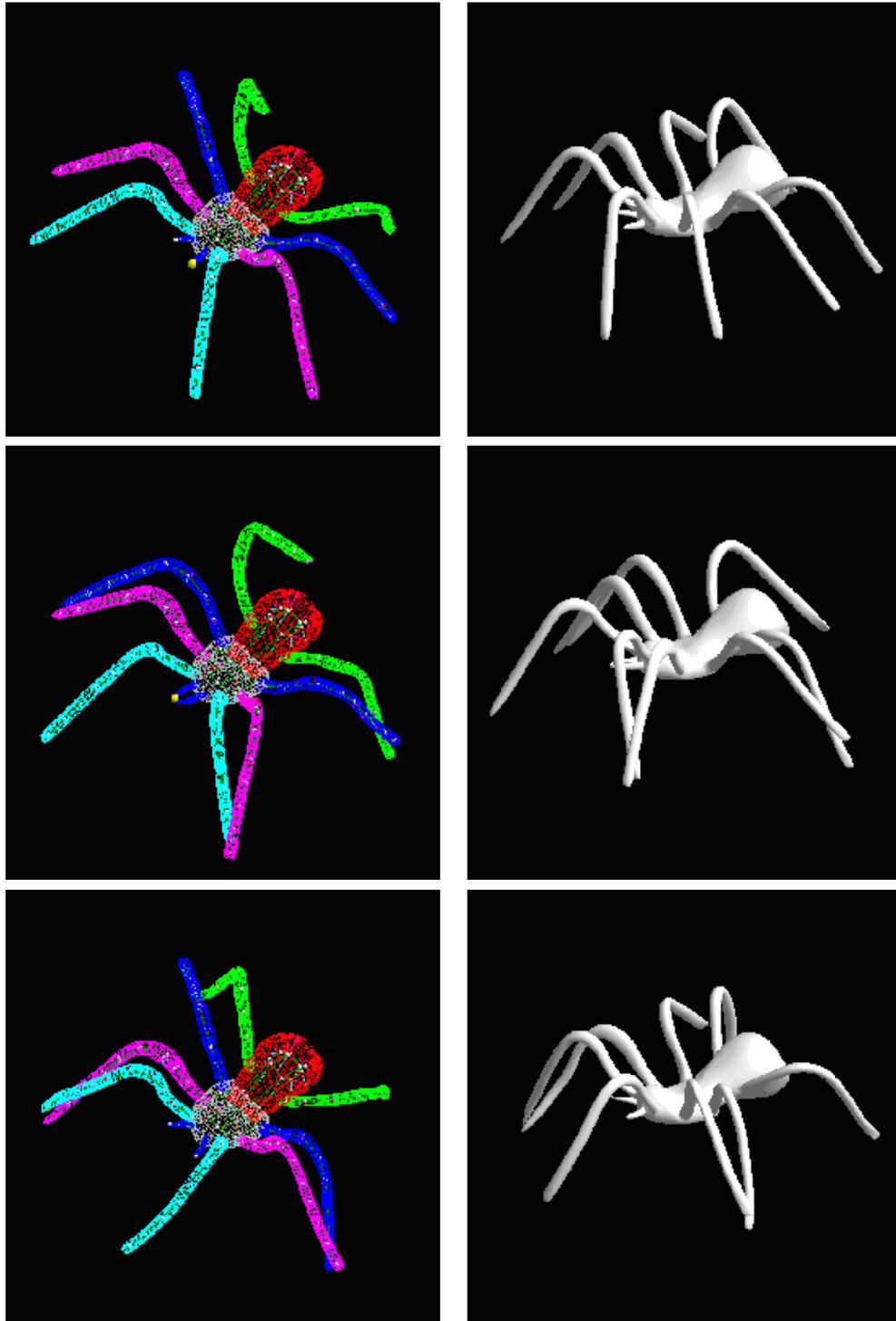


Figure 6.7: The three keyframe m-reps from the spider animation. The top pair show Pose A, the middle pair, Pose B, and the bottom pair, Pose C. The interpolating animation sequence went  $A \rightarrow B \rightarrow A \rightarrow C \rightarrow A \rightarrow B \dots$ .

# Chapter 7

## Discussion and Future Work

Preceding chapters described the genesis of discrete m-reps—the motivation behind their creation and the data-structures and algorithms developed for their use in prototype applications. They discussed past and current implementations of m-reps that demonstrate m-rep modeling and deformation and illustrate the utility of m-reps as multiscale, multifigure shape descriptors.

The premise of this research was that medial representations could be developed into robust shape descriptors having the following characteristics:

- precision based on necessary tolerance, with a notion of intrinsic scale and modeling aperture;
- medial attributes to give better reflection of global geometry than boundary primitives;
- boundary attributes to give better reflection of fine-scale, local geometry than medial primitives;
- object-based deformability;
- hierarchical organization based on object-shape;
- a framework for incorporating statistical shape variations into a model.

These goals have been achieved. Discrete m-rep models provide both local and object-level deformability in multifigure representations and explicitly include object-scale, tolerance and hierarchical level-of-detail. M-reps combine the solid modeling capabilities of constructive solid geometry with the flexibility of traditional b-reps, to which they add multiscale medial and boundary deformations parameterized by an

object-based coordinate system. They provide a tolerance-based skeletal technique that avoids the instabilities of Blum/Voronoi-based skeletal methods; they have been used in applications exploiting these properties for shape design, image segmentation and statistical analysis, and physically based deformation.

This chapter will discuss the tools and modeling paradigms appropriate for multi-scale, multifigure modeling using m-reps. It will then discuss potential and necessary improvements in m-rep technology and will explore promising directions for future research and applications in medical imaging and in computer graphics and computer-aided design. It will conclude with some personal thoughts on the achievements of this work.

## 7.1 Tools and paradigms for multiscale, multifigure modeling

In the course of this work, many insights were gained into the problems of modeling with tolerance-based medial primitives, insights that have pointed to new possibilities for modeling methodologies for CAGD and image-analysis. Multiscale representations have unique characteristics and require new modes of user-interaction in order to be used effectively. Insights on m-rep modeling systems can be derived from other 3D skeletal-based modeling systems (e.g., Igarashi's sketch-based Teddy interface [84] and Storti's research [142], as were discussed in Ch. 2), from implicit surface and convolution surface modeling, and from multiresolution mesh research.

Applications for shape-design or shape-analysis using discrete m-reps share many of the needs of traditional modeling and analysis tools, benefiting from many of the same modes of interaction for modeling and deformation, but modeling using discrete m-reps has the potential for novel interaction techniques, as well. This discussion will address four aspects of modeling with m-reps:

1. medially based object deformation and tolerance-based boundary deformation, with legality checks on acceptable perturbations;
2. construction of multifigure DAGs and tree hierarchies for pseudo-CSG, including considerations of representational polymorphism;
3. blobby modeling and virtual clay construction paradigms;

4. precision modeling paradigms for CAD and other applications, including the fitting of m-reps to b-reps or other shape templates;

This section will discuss algorithmic and user-interface issues involved in implementing these methods in dedicated m-rep modeling systems. A guiding principle for this discussion will be *representational transparency*—that a user of a modeling system should gain the benefits of a particular representation without that representation being intrusive. As Ch. 6 made clear, both Rakshasa and Pablo lack this transparency in their modeling interfaces—medial deformations require explicit manipulation of medial atoms, requiring extensive experience by users to create desired shapes. Rakshasa did include rudimentary operations for *propagated* changes, allowing changes at an atom to automatically modify the rest of the mesh to produce object-based deformations; such techniques will be generalized below.

### 7.1.1 Medially based object deformation and tolerance-based boundary deformation

Shape design systems using m-reps have many of the same needs as b-rep-based systems, and there can be considerable overlap in basic user-interface methodology. It is a frequently heard complaint, however, that the most common 3D computer modeling systems are unintuitive and difficult to master, and this discussion will explore how advantages of multiscale medial primitives can be reflected in improvements to current art in modeling and CAGD. It is my opinion that the problems inherent in most current modeling systems result from two factors: the lack of tolerance in modeling primitives, necessitating overprecision at the wrong stages of modeling; and the lack of representational transparency—when the modeling primitives themselves *get in the user's way*, due to the necessity of paying too much attention to the accidents of the representation rather than the essential shape properties of the object being modeled.

M-reps as implemented in Pablo and Rakshasa have addressed the first issue but not the second—modeling by manipulation of medial atoms in a mesh is no more intuitive than modeling by dragging around spline control-points. Just as manipulating control points for splines is an awkward way to do surface modeling, so is medial atom manipulation an awkward way to do solid modeling. I propose that the sampled medial representation should not be visible to the user; rather, there should be intuitive, solid sculpting tools in place of the atom-by-atom hackwork of Pablo.

What is needed are controls using the medial meshes as underlying structures but not as directly manipulated primitives at the interface level.

One experimental modeling system that addresses representational transparency is Teddy; its use of a sketch-based paradigm for creating figures with extrusions, indentations, and cutting operations allows users to construct shapes with little or no knowledge of the underlying data structures. Its use of a screen-space sketchpad paradigm and of object-based construction—for example, protrusions are created normal to object boundaries—illustrate some useful interaction mechanisms, and the underlying representation is nigh invisible. Teddy also allows object warping based on correspondence between hand-drawn contours. A drawback of Teddy is the lack of mechanisms for precise modeling—a hastily scribbled ovoid may be a blob, but perhaps the user wanted a sphere or other precise shape.

M-reps are well-suited to addressing this last problem, with their separation of global shape properties such as bending and elongation from fine-scale structure at boundaries. It would be almost trivial to rewrite Teddy using m-reps, resulting in a program with similar representational transparency but also having the abilities afforded by the explicit medial structure, the figural-coordinate-based subfigure attachment, and the fine-scale boundary perturbations allowing precise fitting of m-reps to desired boundary positions (see Sec. 7.1.4 below). While a sketchpad paradigm is not ideal for all applications, it illustrates nicely how the data representations can be kept out of the user's way.

### **Medially based object deformation**

Skeletal methods have been used for shape deformation in computer graphics and animation since the mid-1970s, and boundary deformation and object morphing guided by skeletal deformation is well-established in current art. M-reps are heir to these skeletal manipulation techniques and benefit from the intrinsic skeletal structure, which makes unnecessary any processing of a b-rep to establish correspondence with a skeleton created after the fact.

In addition to common techniques for skeletal deformation, Sec. 6.1 noted that m-reps allow medial deformation to be made relative to the medial structure itself; not only are medial-to-boundary correspondences intrinsic, but object deformations may be relative to the medial surface itself, using the medial-atom-based coordinate systems as transported across the surface. This allows deformation to be object-based, rather than based on an external coordinate system or imposed axes. One

way to accomplish this is to compute a parallel transport function between atoms at all positions of an initial mesh before deformation, and then to use this function to compute propagations of deformations. This was approximated in Rakshasa for slice-constrained quadmeshes but not implemented for general mesh structures.

The usefulness of applying deformations at varying scales creates some special requirements, similar to those needed for multiresolution mesh editing. The need to produce bends and bulges at varying scales and the desire for representational transparency in operations leads to interaction paradigms such as those for virtual sculpting or those developed by John Rhoades in his work on bending operators for curved-surface deformation [127]. The use of 3D Gaussian bending operators as shaping tools is one approach.

The creation of medial operators for object-based deformations requires other mechanisms as well. In keeping with the desire for representational transparency, such operations should not be based on the user directly manipulating the medial structure, but rather through the use of bending, stretching, twisting, and bulging or narrowing operators, which are directly interpretable into medial atom and mesh modifications. Operators might be specified by indicating surface regions, giving a corresponding medial operating region based on  $(u, v, t) \mapsto (u, v)$ , or by mechanisms to specify solid regions of the object, e.g., "I want to grasp the object here and here and *pull* it or *twist* it or make it *bulge* or *narrow*."

### **Tolerance-based boundary deformation**

Sec. 5.1.3 discussed some basic issues of using approximated width values in boundary displacements, based on interpolated  $r$  on the subdivision boundary. This is typical of the way an m-rep modeling system uses the medial tolerance information provided by the medial atom mesh to specify shape characteristics. Interactive modeling should allow the boundary to be modified by a variety of tools; again, virtual sculpture and virtual painting paradigms are a good approach.

For boundary displacements based on displacement mapping, interactive editing will need to interactively modify displacement maps, mapping from surface coordinates to displacement texture space in the same way that current 3D paint packages allow polygonal, spline, or subdivision models to be painted by a virtual brush. These displacement brushes will provide protrusion or indentation operators of varying shapes and may be applied by projection of 2D Gaussian scalar fields to the surface in medially based coordinates.

For boundary displacements based on displacement meshing, techniques for multi-resolution mesh editing are appropriate, allowing shaping tools to be applied at any subdivision depth and changes to be propagated to other mesh levels. The interpolating nature of the m-rep IIS-surfaces makes this somewhat simpler, in fact. In keeping with the goal of representational transparency, changes in subdivision depth should be invisible to the user; broad changes over large areas will be applied coarsely and interpolated to the finer levels, while detailing will be applied locally to the fine-scale meshes.

Tools may be applied to the surface as  $3D$  Gaussian scalar fields of varying shape and orientation, or by  $2D$  ones as per the displacement mapping case, with larger ones being applied to the coarse mesh and smaller more locally to the finer. (Such tools may be used for medial bending operators as well.) The switching between fine and coarse level meshes can be used as an adaptive refinement mechanism, using the coarser levels for medial modifications and broad boundary perturbations and then subdividing more finely for finer detail.

In keeping with the idea of m-reps as solid modeling primitives (rather than as glorified b-reps with built-in skeleta), boundary displacements may be propagated along  $\tau < 0$  on the interior of the m-rep, thus warping the interior of the model. This has ramifications when subfigures or other objects are anchored in the interior by  $(u, v, t, \tau)$  positions. This will be discussed below in Sec. 7.1.3 and Sec. 7.2.3.

### **Legality checks on m-reps during interactive modeling**

The previous chapter discussed legality checking in Pablo, which was used to prevent boundary self-intersections during automatic model-fitting during segmentation. Similar legality checks can be employed in a interactive m-rep design tool, restricting the allowed deformations or signaling the user when a model has been unacceptably perturbed. Although the user should rarely need to manipulate medial atoms directly, the interface should detect, for instance, when medial bends become too sharp and the boundary would pleat or self-intersect. Another area where legality checking is necessary to keep the displacements at the boundary within width-proportional tolerances and to detect whether medial operations have made previously legal boundary perturbations no longer acceptable.

A modeling system with representational transparency will not simply detect these occurrence but deal with them. Thus, an m-rep model should either resist such a change or alter itself to avoid the boundary pleating. If a bulge exceeds curvature

tolerances, for instance, an editing system might inform the user that a subfigure should be attached. In an intuitive sense, this is akin to sculptor needing to add a blob of clay to a figure, rather than trying to pinch out an extrusion from the main figure. A more advanced system would automatically substitute a subfigure in the region of the previous boundary deformation—this would be more in keeping with the desired representational transparency.

### 7.1.2 Multifigure DAGs and tree hierarchies for pseudo-CSG

Pablo has had the ability to construct multifigure models since the early days of its inception, but the use of these was limited by the difficulty of creating smoothly surfaced boundaries for them. With the development of the boundary remeshing scheme for attaching protrusion and indentation figures, m-reps have only now become capable of general modeling tasks involving multifigure DAGs and tree hierarchies.

A modeling system for creating multifigure DAGS and tree hierarchies will probably use a deformable CSG paradigm, adding or subtracting subfigures from a main figure and having tools to manipulate the subfigures and fit them into place. Interfaces might be as informal as a sketch-based modeling system or as precise as a typical CAD system.

Another design paradigm takes a shape morphogenesis approach, creating boundary bulges or dimples that grow and become separate subfigure protrusions or indentations. A multifigure modeling system must also take into account considerations of representational polymorphism. A good modeling system should also provide linear and loop figures, based on sampled  $1D$  medial loci. Such  $1D$  medial axes are non-generic as  $3D$  Blum primitives, since boundary perturbations cause the formation of sheetlike axis regions, but they are fully generic in tolerance-based m-reps, where fine-scale perturbations are captured by the boundary displacements.

### 7.1.3 Blobby modeling and virtual clay modeling paradigms

With the ability to remesh subdivision surfaces to produce both indentation and protrusion figures, and with the existence of finite-element methods for physically based deformable modeling, m-reps are well-suited to virtual sculpting applications. Such applications are another way to provide representational transparency, by treating modeling primitives as surfaces or solids to be modified using haptic (force-feedback based) or haptic-mimicking paradigms, treating the solids as virtual clay. Such sys-

tems generally use subdivision surfaces as multiresolution boundaries; another approach, that of McDonnell [103], uses subdivision solids, a 3D subdivision meshing with physically based properties. M-reps would fit well into either type of system, and add medial deformability to the object-modeling process.

Given straightforward improvements to the remeshed joining of figures described in Sec. 5.2.3, m-reps may be combined in CSG fashion with user-tunable filleting of join regions. This will let them be used as blobby modeling primitives, similar to convolution surfaces, allowing medially defined surfaces to be joined simply by sticking them together.

One volume-based aspect of m-reps relates to their parameterization of the figural volume. As mentioned in Sec. 7.1.1, not only is the m-rep interior (and near-surface exterior) parameterized by  $(u, v, t, \tau)$ , but the displacements on the boundary can be extended parametrically to the volume of the object based on this figural correspondence. Thus, an m-rep may provide a solid modeling of an object with imbedded indentation subfigures or organelles.<sup>1</sup> This will be discussed further in Sec. 7.2.3.

#### 7.1.4 Precision modeling paradigms

Although m-reps as used in medical image modeling look like blobby primitives, this appearance is deceptive; the medial meshes define the boundary to within a tolerance, but the boundary itself can be precisely positioned by the displacements on the subdivision surface. For m-reps to be useful in CAD systems, they will require a broad array of shaping tools for placing boundaries, edges, corners, and other high-frequency detail in the displacement-meshed boundary. In keeping with the desire for representational transparency, such operations should not require knowledge of the particular subdivision mesh structure (whether IIS-surfaces or perhaps adaptive, multiresolution meshes, see Sec. 7.2.1) but should provide mechanisms for indicating, with necessary precision and tolerance, the shape desired for the object.

M-reps already can automatically fit to 3D image data; needed as well are figure-based methods to fit m-reps to b-rep models, with automatic perturbation of boundaries and indications where boundary tolerance is exceeded and a subfigure is necessary. b-rep to m-rep approaches can be generalized to geometric-shape-templates defined either as b-reps or by explicit or implicit functions; m-reps can be fit to implicit functions by the same methods used for image-gradient-based segmentation; they can

---

<sup>1</sup>like bugs in Jello, extending the work of Heckbert [79] on *dessert tracing*.

be fit to explicit function boundaries using raycasting along normals or perhaps using approaches drawn from active-surface modeling or from Pablo-style segmentation, generating subfigures as necessary to prevent beyond-tolerance deformations.

In this use of m-reps for *shape capture*, they serve as figural templates; just as m-reps are well-suited to atlas-driven segmentation, they will be well-adapted to a shape-catalog approach to modeling, allowing both the fitting of m-rep models to specific b-reps (or other intolerant representations) and the figure-based perturbation of such models.

For precision modeling, m-rep boundaries must be able to provide sharp, precise edges, creases and corners. Edges—the  $\mathcal{C}^0$  meeting of two locally planar surfaces—are in a sense “natural” to m-rep crest regions, corresponding to the outer edges of the medial mesh, and various methods can be devised to create edges of varying sharpness in these regions. Corners—the  $\mathcal{C}^0$  meeting of three locally planar surfaces—are perhaps less “natural,” but might still be dealt with on the basis of placing edges and corners along  $(u, v, t)$ -parameterized curves of constant  $u = u_{\min}$  or  $u_{\max}$ ,  $v = v_{\min}$  or  $v_{\max}$  and  $t = 1, 0$  or  $-1$  in crest regions.

The more general cases of placing edges, corners and creases on surfaces must rely on properties of the displacement subdivision boundaries. Sec. 4.1.7 discussed the difficulty of applying some of the standard corner- and crease-producing algorithms used for Catmull-Clark surfaces to the interpolating IIS-surfaces implemented for m-reps. The use of boundary displacements, as described in Sec. 5.1.2, provides a partial solution to this problem but, in using either displacement maps or meshes, is restricted to the precision of the map-resolution in  $(u, v, t)$  or the fineness of the finest-scale displacement mesh to specify the precision of the edge or corner. A mechanism is necessary for analytically specifying an edge or corner in object-based  $(u, v, t)$  coordinates and requiring the displacement surface to conform to it at any subdivision level. Similarly, smoothness and flatness are also fine-scale properties of a surface and should be specifiable in the same way as other fine-scale surface details.

### 7.1.5 Last remarks on m-rep modeling methodology

As is clear from this discussion, m-reps bring to bear powerful new paradigms for shape design. This discussion has been speculative in nature, and the actual development of m-rep interfaces will be best served by close collaboration between end-users, familiar with the needs in their particular fields and their existing design tools, and m-rep technologists, armed with multiscale, multifigure primitives and wielding the

concepts of representational tolerance, polymorphism, and transparency.

## 7.2 Improvements to current m-rep technology

M-reps are a work in progress. As applications for them expand, as considerations for modeling interfaces such as those described above are taken into account, and as the needs for further development in Pablo are made clear, m-reps will change and advance. This section discusses some of the areas where attention should be directed to improve and augment current m-rep technology with abilities needed for future applications.

### 7.2.1 Better m-rep boundary surfaces

The current IIS-surfaces give a quick and simple fit to the m-rep boundaries, but, as was discussed in Sec. 6.2.3, they suffer from rippling effects due to the cubic-spline-like effects of the underlying Catmull-Clark algorithm applied to the coarse involute mesh; this is especially evident when normal interpolation is active with a small tolerance on  $\theta$ -deviation. Such rippling can be eliminated on an *ad hoc* basis by adjusting  $\eta$  at crests and boundary perturbations across the surface; still, it introduces undesirable fine-scale boundary information, particularly as magnified in perturbed boundary normals.

A better solution is to use a more advanced multiresolution surface fitting providing control of boundary curvature. Methods described earlier by Halstead [77] or Litke [97] seem well-suited to m-rep boundaries; in general, multiresolution mesh techniques seem a good fit with m-rep technology and should be explored in future research. Of particular interest are systems allowing adaptive subdivision for regions of higher boundary curvature.

### 7.2.2 Improvements in boundary-parameter interpolation

As a separate but related area to the above is the general problem of creating and interpolating a medial-based parameterization on the boundary, as well as interpolating the  $\hat{r}$  field approximating the associated medial  $r$  for the boundary positions. Such interpolations are required across tiles at coarse subdivision levels as well as in the subdivision process itself to establish values for the finer-level meshes.

The current interpolation methods use bilinear averaging between mesh levels and quad-splitting and bilinear interpolation for a quad at a given level. The quad-splitting lends an arbitrariness to the interpolation across non-planar tiles; a better approach might be to fit spline patches to non-extraordinary subdivision regions to give a 1-1 correspondence of patch and  $(u, v, t)$  parameters and perhaps to use Stam interpolation near extraordinary vertices. In regard to  $\hat{r}$ , a better approach would use a constrained spline fit, interpolating  $\hat{r}$  for known involute  $r$  values and producing a  $\mathcal{C}^2$  continuous  $\hat{r}$  field elsewhere. While the IIS-surface algorithm could do this interpolation, it would not prevent ripples in the  $\hat{r}$  field. Better theoretical insights for this may be found in the work of Damon, but one should be leery of taking too *Blum* an approach and not making use of the tolerance allowed in approximating  $\hat{r}$ .

Interpolation issues become especially problematic in figure-subfigure join-regions, with a separate surface parameterization from either parent or child  $(u, v, t)$  and with  $\hat{r}$  and boundary normal  $\vec{n}$  varying very rapidly in shifting between figure and subfigure width. Although  $\vec{n}$  can be computed by limit masks for the remeshed subdivision boundary, an  $\hat{r}$  interpolated by bilinear interpolation over the new grid vertices (as is standard with Pablo) loses the information from a more continuous, Blum-like description of  $r$  and  $\nabla r$  in a join region.

### 7.2.3 More systematic boundary and internal displacements

Boundary displacements can also benefit from multiresolution mesh based approaches. The displaced meshes currently in place were created as a straightforward prototype implementation. They work well within the IIS-surface framework but seem a bit naïve when compared to state-of-the-art work. A perusal of the latest SIGGRAPH Proceedings reveals a wealth of research in displacement texturing methods from which multiresolution schemes might be drawn for use on m-rep subdivision boundaries. In any case, boundary displacements need to be more fully integrated into the current system, with computation of correct boundary normals for the displaced meshes. An approach such as that of Desbrun et al. [44] could provide normals and higher order approximations to differential geometric properties for the perturbed meshes.

As previously stated, displacement meshes can eliminate the surface-rippling associated with interpolating subdivision surfaces; manual smoothing tools could allow selection of surface regions to be smoothed as necessary, or automatic curvature/energy minimization techniques could be used selectively or over the whole surface.

Because medial mesh resampling has not yet been discussed, one problem not yet addressed is that of altering displacement fields or meshes for a new boundary generated by an m-rep with a new, interpolated row or column of atoms. The problem of distance-field resampling during interactive object modification requires careful attention. One approach to remapping boundary coordinates is to use diffeomorphisms on the medial implied boundaries to establish correspondences between old and new  $(u, v, t)$  coordinate locations. This was used by Crouch in her FEM research [33]. In any case, m-reps require better interfaces for creating and modifying displacements, based on the ideas discussed in Sec. 7.1.

As noted in the previous section, it is an error to think of m-rep boundary perturbations as simply changes in the location of the boundary itself; the term *boundary displacement* gives a false impression and is perhaps a misnomer. Because of the parametric correspondence between internal and boundary locations, perturbations of a boundary position at  $(u, v, t)$  by some  $\phi\vec{n}$  must be thought of as propagating to other parametric positions  $(u, v, t, \tau)$  by  $\phi(\tau)\vec{n}$ , where  $\phi(\tau)$  smoothly decreases with distance from the boundary at  $\tau = 0$ . Thus, what is produced is a warping of the interior space of the m-rep figure. This becomes important when the solid body properties of the m-rep come into play, such as in positioning objects within a figure, as will be discussed below in Sec. 7.3.

## 7.2.4 Improved interfaces for multigure design and medial modifications

Pablo provides the only current shape-design interface for m-rep modeling and has drawbacks as previously discussed. New shape modeling tools are necessary, incorporating many of the techniques discussed in Sec. 7.1. In particular, better tools need to be implemented for multigure model-creation and for shape modifications in a representationally transparent fashion.

Multigure models have certainly been created, even in the early work of Chen, which showed a kidney with an indentation figure, and by Fletcher in SCAMP with the deformable hand models. Multigure Pablo models include a two-figure liver model, a three-figure prostate with seminal vesicles, the lateral brain ventricle of Fig. 6.4 and models of heart chambers and blood vessels currently being build by Pilgram et al. (see Sec. 6.3.1). With the new implementation of remeshed figure-subfigure joins, new design tools should be created allowing model creation to include the varying

filleting of such join region and, as mentioned above, parameterization issues in the join regions need to be addressed.

One area of current research is on protrusion subfigures and internal, unattached subfigures within a parent figure. In keeping with the use of figure-based coordinates, the locations and behavior of these internal subfigures are tied to the medially defined coordinates of the parent so that object-based deformations and boundary-perturbations on the parent are reflected in the subfigures' shape and location. Thus, an indentation figure is anchored not simply at boundary hinge atoms but at all internal atoms, each of which will be positioned at  $(u, v, t, \tau)$  locations in parent-figure coordinates. Based on these coordinates, bending and elongating medial deformation of the parent can be propagated to the child figures; changes in parent figure  $r$  values can also be propagated to the child figure's medial atoms, along with displacements of the boundary. Tools to support these multifigure operations in a representationally transparent fashion would be an important addition to Pablo functionality.

### 7.2.5 Trimeshed boundaries and new parameterizations

M-rep boundaries are implemented as quadmeshes, based on the initial mesh of involute positions and subdivided at each level by the modified Catmull-Clark algorithm. There are advantages that might be gained from using a trimesh rather than a quadmesh structure. Seurat routines currently splits quads into tris for many purposes—e.g., Phong-normal interpolation,  $(u, v, t)$  interpolation and near-point computation—and the code might be simplified by going to a trimesh with interpolating Loop scheme, using a similar IIS-surface algorithm as that implemented for Catmull-Clark surfaces. While the Catmull-Clark surfaces handle generating meshes containing triangles, the obvious quad-splitting to produce a consistent trimesh structure for an entire m-rep will leave most vertices of valence-6 and therefore with bad curvature.

Trimeshes will have obvious advantages, e.g., allowing adaptive subdivision depths dependent on curvature in a much more easily implemented fashion. There may be difficulties with a less obvious  $(u, v, t)$  correspondence, but one can do a similar interpolating subdivision of the tri-boundary's parameter space as is done for current quadmeshes, holding initial  $(u, v, t)$  coordinates constant at corresponding vertices between levels.

### 7.2.6 Trimeshed medial axes and interactive resampling

Along with a trimeshed structure for the boundary subdivision, there are perhaps even greater advantages to a a trimesh structure for the mesh of medial atoms. A medial trimesh would allow interactively resampling of the medial mesh during deformation, in order to maintain an adequate medial sampling. Quadmesh-based m-reps could only do this by adding an entire row or column of medial atoms to the mesh; a triangulated medial mesh would allow individual medial atoms to be added to the mesh in a small region subject to elongations or other distortions of the implicit medial structure. Techniques could also be devised as well for reducing the sampling frequency of the mesh.

There are a number of ways the medial mesh might be resampled. New medial atoms might be interpolated by the boundary-based methods discussed in Sec. 4.3, based on boundary  $(u, v, t)$  positions corresponding to the approximate medial  $(u, v)$ . Alternately, advances in the interpolation of medial atoms, described below, might be used to interpolate new medial atoms directly. Medial-atom averaging schemes might be used to reduce the sampling density in a trimesh.

Another approach to this might be to produce discrete m-rep figures by resampling using Paul Yushkevich's continuous m-reps. One might fit a cm-rep to a given m-rep figure—since cm-reps work quite well for single-figures if they are Blum, and then resample the entire surface at a different density, perhaps using point-repulsion on the medial surface followed by Delaunay triangulation. Such a point-repulsion algorithm might be modulated by the  $r$ -field on the cm-rep to vary the sampling density inversely with object-width.

The creation of new medial atoms or the removal of old ones will alter the  $(u, v) \Leftrightarrow (u, v, t)$  correspondence between medial and boundary parameter spaces. This is also the case when adding new rows or columns to a quadmesh. This is undesirable with regard to any statistics which have been gathered based on these coordinates. As well, interpolating displacement maps or vertex-displacements based on a resampled boundary is also problematic. One approach to this problem would involve a normalized  $(u, v, t)$  with coordinates in the range  $0 \leq u, v \leq 1$ , though this will eliminate the feature of  $(u, v)$  being themselves an approximate width-proportional distance function on the surface, based on the sampling density. A different approach would be to parameterize by a medial-axis-of-medial-surface method, parameterizing the medial surface along its medial axis and transversely in the involute directions. In this way, the coordinates would be tied to the geometry of the medial surface, rather

than to any particular sampling of it. These approaches are all speculative but point to fruitful areas for research.

### 7.2.7 Fitting m-reps to b-reps or objects in volumetric data

The fitting of m-reps to b-rep models is still in its infancy, with only a few experiments having been done using polyhedral templates to produce boundary displacements. Certainly if m-reps are to be used as modeling primitives for computer graphics, there must be reliable means to transfer models from standard b-rep format to m-reps.

One need for this, discussed above in Sec. 7.1.4, is to give m-reps the ability to model precise objects. A polyhedral-template approach will allow an m-rep to be given precise boundary perturbations. Atlas-based matching approaches are also conceivable; e.g., a 4-legged-mammal m-rep could be created and then fit to various b-rep models in a semi-automatic fashion. Automatic model-fitting could use a similar paradigm to that created for *3D* image segmentation: a similarity transform for a rough fit, then medial atom transformations, then perturbations of the boundary in normal directions to intersect the boundary. This would use the same coarse-to-fine figural approach developed for segmentation, fitting main figures, then indentation and protrusion figures, as well as unattached *satellite* subfigures either internal or external to the figure. Such a method will show the strength of a figural, atlas-driven approach for extracting semantic, structural information from bare b-rep models.

This could work with current technology, performing a *3D* scan-conversion of the b-rep into a volume and then using Pablo to do the segmentation based on an m-rep model. Or, one could analyze a whole collection of objects, as per Styner, derive an m-rep that captures the medial structure and variability of the different objects, then use this m-rep for “figural segmentation” of similar b-reps. (Speed issues might present problems here; Styner’s m-rep training methods were slow.) Automatic creation of an m-rep for an unknown b-rep could involve an initial skeletonization and pruning followed by finding of an m-rep giving a best fit. Once the m-rep structure—topology, number and location of subfigures—has been decided on, automatic fitting techniques can be applied to give a precise match. The skeletonization method used by Styner (that of Attali and Montanvert [3]) would probably be replaced by the better Hamilton-Jacobi skeletonization technique of Siddiqi et al. [136]. This technique is superior to Voronoi-based methods in many ways, including allowing a smoothing parameter to eliminate fine-scale noise and producing a Blum axis with full medial information  $(\mathbf{p}, r, F, \theta)$  (see Dimitrov et al. [45]).

### 7.2.8 Improvements in medial techniques

Mathematical analysis has enhanced our understanding of Blum and related medial axes. With this new understanding comes new abilities.

Crouch produced good medial interpolation surfaces for single-figure m-reps, and there has been recent work by Michael Kerckhove at the University of Richmond on a better medial interpolation. Rakshasa and Pablo abandoned medial interpolation from a coarse mesh and moved instead to a boundary interpolation based on the coarse medial sampling. Kerckhove has gone back to the idea of interpolating the medial sheet directly and has developed an interpolation over a 10-manifold with 2 constraints that gives reasonable medial atoms to interpolate  $(\mathbf{p}, r, F, \theta)$ . Given this ability, one could then do on the fly resampling and medial atom interpolation to produce new medial meshes which retain a desired sampling density. Also needed would be the inverse operation, replacing medial atoms in a contracted region by fewer “averaged” medial atoms.

Cm-reps are another method that provides a continuous model for the medial surface, as noted above in Sec. 7.2.6, and cm-reps might be applied to many applications in conjunction with or as alternatives to discrete m-reps. In any case, they provide an experimental tool for studying computational aspects of Blum medial objects with  $\mathcal{C}^2$  medial axes; because of this, cm-reps are worthy of further research in the medial shape analysis and modeling community.

### 7.2.9 Improvements in corner and edge specification

Sec. 7.1.4 discussed the need for precise corner, crease and edge creation for CAGD application of m-reps. While Sec. 7.2.1 and Sec. 7.2.3 discussed needed improvements to boundaries and boundary displacements, they did not discuss the corner issue explicitly. Although the technical aspects of adding sharp corners and edges to m-rep models is well-addressed by the multiresolution surface approaches, the theoretical basis for incorporating edges and corners into an m-rep framework requires further attention. As an example, the corners of a brick, though describable by boundary displacements from an ellipsoidal blob, nonetheless contain coarse scale location and orientation information relative to the medial structure of the blob. There is currently no systematic approach to treating corners and edges as coarse-scale features of an m-rep; they are at present simply artifacts of the boundary perturbations. What are needed are explicit specifications of these special points and loci as coarse-scale

structures with medially parameterized locations.

Theoretically, insights might be drawn if we see this as a shift from a scale-space based on a Gaussian blurring function to one based on a variable conductance diffusion approach, as was discussed in Ch. 2 in Sec. 2.3. In this way, an anisotropically blurred brick, as per Peron and Malik, retains an inherent “brickness” even at coarse scale, distinguishing it from blurred ovoids or footballs.

Practically, this might simply require specification of corner and edge features in medially defined coordinates on a figure-by-figure basis for an m-rep, with the number and orientation of these features being included in the coarse description of the object provided by the m-rep. As an alternative to this, edges and corners might instead be described relative to the crest regions of the medial sheet. In either case, a corner or edge will have a coarse-scale description tied to the figure as well as a fine-scale description as a boundary perturbation. Similarly, smooth and flat regions of a surface may have both coarse and fine-scale descriptions.

## 7.3 Directions for future m-rep applications

While the above are areas where research can be directed to improve m-rep technology, this section will discuss how m-reps can be applied to research problems and applications.

### 7.3.1 M-reps in medical image analysis

With m-reps now being used for a variety of segmentation, registration, and shape-analysis tasks, one of the main objectives in their medical application is to provide primitive elements for atlas-based image-segmentation. These atlases will include statistics on expected variability of an organ, variations over a single m-rep, as well as on shape-morphologies—the presence or absence of figures or subfigures producing different m-rep representations. M-reps have also been applied as shape-descriptors in discrimination studies.

M-reps have strengths and weaknesses as geometric primitives for image analysis. This section will first discuss the *correspondence problem*, then discuss applications of m-reps to deformable solid modeling, to 3D data analysis and display, and to stochastic model creation.

## Addressing the correspondence problem

Attempts to use the  $(u, v, t)$  coordinates for boundary correspondence suffer due to the non-uniqueness of medial sampling—objects might have identical boundaries but very different  $(u, v, t)$  parameterizations. Peculiar medial sheets can produce fine-looking objects—in particular,  $(u, v, t)$  can be skewed badly on the boundary, though boundary itself is well-behaved. This is true even for simple, single-figure models; representational polymorphism makes more complex objects even more problematic. The work of Greg Clary [30] illustrated this problem; in his work on the analysis of gated cardiac images, he found that early m-reps could accurately segment the heart data but not maintain correspondence across a time-series. Crouch solved this problem for m-reps in her FEM deformation research, using diffeomorphisms over boundary parameterizations to establish geometric correspondences, as discussed in Sec. 7.2.3.

Another way to address the correspondence problem is seen in the work of Davies et al. [42], who establish boundary correspondences automatically by recasting the problem to one of find an “optimal” parameterization of the shapes in their training sets—*optimal* in the sense of providing the tightest statistical distribution over the training data. New approaches in Pablo have employed the principal geodesic analysis of Fletcher so that, rather than basing optimizations on correspondences between  $(u, v, t, \tau)$  coordinates, the Mahalanobis distance based on  $\log p$  of the medial structure itself is used. Application of Davies’ automatic methods in m-rep research will use the variance over the medial Lie groups as the statistical measure, automatically establishing shape- and volume-based correspondences. A shift of emphasis is evident in m-rep-based correspondence research; surface models by their nature require correspondences between relative surface points, whereas m-reps, which are *volume primitives*, are better suited to and better served by correspondences between interior positions or regions, incorporating  $r$ -proportionality according to the multiscale nature of m-rep models. M-reps will also require methods for establishing correspondences for special positions in respective model interiors for specific applications; these special places may be geometric, structural or functional landmarks.

## Better deformable solid modeling of tissues and organs

As described above in Sec. 7.2, an m-rep’s  $(u, v, t, \tau)$  figural coordinates parameterize its solid volume. Thus, changes in the medial or boundary structure are reflected on

the volume of the object, not simply the boundary. This allows m-reps to model the behavior of deformable solids using manual, kinematic, physically based or statistically based deformations of the multigure structures. These could be integrated into Pablo-like systems and would expand the capabilities of m-reps as a deformable solid representation.

With regard to physically based models, the work of Crouch on FEM using m-reps for tissue deformation holds promise. Modeling tissue deformation is an active area of research with applications in image-registration for surgical planning, in surgical simulation and in forensic analysis. Crouch's methods did not include indentation subfigures. In some cases, m-rep polymorphism might allow this problem to be side-stepped; in others, a different remeshing scheme might be in order; Crouch speculates that a tetrahedral rather than hexahedral remeshing scheme would make algorithms for indentation subfigures more approachable [34].

Also useful for physical simulation would be tests for self-intersection for models under deformation. Non-local self-intersection, where a figure interpenetrates with a subfigure or with itself, could be tested using modified code for computing the  $(u, v, t, \tau)$  coordinates for a point in space—a self-intersection occurs when a point on the boundary has a negative  $\tau$  value relative to another medial region of the object. (Local self-intersection is detected already by checks on surface folding.) Coming up with a fast and robust multiscale test for m-rep interpenetration presents an interesting research project.

There are many possibilities presented by the  $(u, v, t, \tau)$  coordinate system. Promisingly, figure-based coordinates might be extended from the description of subfigure positioning within or near a parent figure to the use of multiple coordinate systems to describe positioning of objects within multi-object groupings. This would create hybrid coordinate systems based on positions and orientations relative to multiple figures; research in this area might explore the mathematics necessary for creating accurate multiscale distance functions.

### **Improved interfaces for 3D data analysis and display**

Other improvements to m-rep applications in medical image analysis will come by addressing user-interface and modeling issues raised in Sec. 7.1, especially those that will provide better mechanisms for building, deforming, and computing statistics on complex m-rep DAGs and trees. Such improvements will enhance considerably the usefulness of discrete m-reps as shape descriptors for image-analysis purposes.

M-reps may also drive improvements in modes of display for volume data and atlas-segmented scans. Such improvements might be based on the work of Chen for m-rep-guided volume-rendering or on simply displaying m-rep segmented objects as b-reps with varying degrees of transparency necessary to reveal internal details and figure-based substructure. In either case, the use of atlas-based segmentation by m-reps holds tremendous potential for automatic analysis and display of volumetric image data.

### **Stochastic model creation**

Because m-reps were designed to model *classes* of objects, based on deviation from a stock model, they are suited by their nature to generation of multiple, varied instances derived from an average object. Using statistical priors on medial meshes and boundary displacements and on subfigure positioning (in parent coordinates) and deformation, stochastic models can be used to generate unique instances from a population using a base m-rep. In order to use this in a design application, the statistical modes of variation must be specified according to default parameters, which will depend on medial width and acceptable boundary curvature.

When using m-reps for image-analytic purposes, this process is reversed. Discrete m-reps are fit to individual members of a population of similar objects, and the statistical characteristics of the variation between models is computed; this, then, becomes part of the statistical characteristics of the m-rep. Thus, an m-rep is not simply a collection of geometric primitives, more-or-less isomorphic to any particular b-rep; rather, an m-rep is a statistical model of a class of objects. This view is born out in the Lie group theory and the use of PGA for modeling statistical distributions of shape from m-reps, as discussed in Sec. 6.3.4.

An atlas of m-rep models must also provide not simply varying single m-reps but distributions for topological and geometric variations in m-rep structure—e.g., topologically, the probability of an organ having one or several additional subfigures; geometrically, the permitted variations that avoid illegalities such as interpenetrations between figures and subfigures.

### **7.3.2 M-reps in computer graphics and CAD**

M-reps are uniquely posed to provide representational transparency in modeling systems, with interactions as discussed in Sec. 7.1. They have potential as design prim-

itives in a wide variety of modeling interfaces: in b-rep modeling systems, because m-reps can be converted easily to spline or polygon or subdivision-based b-reps; in solid-modeling editors, because they can perform CSG-style additive and subtractive modeling; in sketch-based design editors, by integrating medial and figural modeling primitives; in haptic-based modeling systems, by combining a multiresolution surface approach with a deformable skeletal structure; in virtual surgery simulation, by their ability to model physically based deformation; in volume graphics systems, as deformable volume primitives.

Based from their conception on the need to do constructive figural modeling with tolerance-based, deformable primitives, m-reps can be applied to CAD and graphics applications that can benefit from both the global deformability and the fine-scale surface deformations. There are a number of areas of promise for m-reps in computer graphics and computer-aided design:

- as skeletal, figural subdivision primitives for design and animation;
- as objects allowing automatic instantiation according to statistical models;
- as CSG-style solid primitives, with tolerance and variable filleting of joins;
- as primitives for medially based LOD decomposition and rendering;
- as primitives for object-based texturing and automatic texture generation;
- as model-based shape-capture primitives for converting from other models or image-based data to a solid, multifigural representation;
- as deformable volume-primitives for volume rendering and volume graphics;
- as multiscale primitives allowing adaptive computation of physically based properties.

This section will briefly discuss each of these.

### **Skeletal, figural subdivision primitives for design and animation**

M-reps combine already established skeletal techniques for object manipulation with techniques for subdivision boundaries and add their inherent figural representation with its object-centered coordinate system. As such, they seem ideal for many general

modeling tasks involving object dynamics, multiscale deformation, key-framed, kinematic and physically based animation, and procedural modeling and texturing. One potential application is as skeletal-based subdivision-bounded primitives for game engines. Their subdivision boundaries provide advantages over polygonal boundaries at negligible rendering-time cost, and the inherent, object-based deformability provided by the medial structure and boundary displacements provides advantages over b-reps such as splines or ordinary subdivision surfaces. In essence, they bring the power of deformable solid modeling into an interactive design or rendering environment.

### **Automatic instantiation of stochastic models**

Because m-reps were designed to model *classes* of objects, based on deviation from a stock model, they are suited by their nature to generation of multiple, varied instances derived from an average object. Using statistical priors on medial meshes and boundary displacements and on subfigure positioning (in parent coordinates) and deformation, stochastic models can be used to generate unique instances from a population based on a base m-rep. In order to use this in a design application, the statistical modes of variation must be specified according to default parameters, which will depend on medial width and acceptable boundary curvature.

In this way, m-reps can be used in systems requiring statistical or procedurally based appearance models, such as those generating crowd-scenes for animations, as well as incorporating statistical deformations for these models on a figural basis.

### **Medial-based LOD decomposition and LOD rendering**

One of the initially conceived purposes of the m-rep was as a new rendering primitive. It was hoped that the multiscale medial paradigm could be leveraged into level-of-detail approaches for rendering complex objects. Because they combine a stable framework for displacement subdivision surfaces with a tolerance-based, medially defined figural structure, m-reps do appear to be well suited to applications requiring LOD decomposition—including topological simplification—of complex figural models.

Section 5.1.2 discussed some general ideas for LOD simplification using displaced IIS-surfaces. In addition to such boundary-based techniques, common to subdivision surface representations, m-reps provide figural representations with object based scale information which might be leveraged to guide object simplification algorithms which include topological simplification, as subfigures are pruned when they fall below a de-

sired resolution. Similarly, boundary perturbations may be smoothed while the figural structure remains unchanged and figure-based deformation remains unhindered.

### **Object-based texturing and automatic texture generation**

M-reps provide solid coordinates  $(u, v, t, \tau)$  with width information for each medial figure; this information may therefore be used to provide texture coordinates for boundary or solid texturing. Because the model also provides width information for each of these coordinates, such information may be used to modulate the texture functions in an object centered fashion that is invariant under similarity transformation and allows the textures to be deformed with the model as per medial and boundary deformations. This could be used for both stock image textures as well as in automatic texture generation schemes such as those of Turk [149, 150] and of Wei and Levoy [154]. For instance, in reaction-diffusion texturing, information on  $r$  and  $\nabla r$  could be used to modulate the reaction for interesting and scale-based effects tied to the object geometry.

### **Shape-capture from other model types or image-based data**

One of the chief uses thus far for m-reps has been for image-segmentation, fitting their multigure models to objects represented by intensities  $3D$  datasets. M-reps bring with them the capability of creating a medially based multigure representation for other models. Sec. 7.1.4 discussed ideas for capturing object shape from b-reps and other representations; such techniques allow models to be imported into an m-rep-based system and to inherit the tolerance-based multiscale deformability that m-reps offer.

Another class of shape information is that derived from image-based information. Oh et al. [110] demonstrated that an IBR representation could be generated manually from  $2D$  photos lacking disparity/depth information, and other methods have been used to create  $3D$  geometric models based on single or multiple  $2D$  images either with or without disparity. M-reps could bring a model-based multigure shape paradigm to bear on geometry extraction from image-data. An object of known topology and figural connectivity could be extracted from a single view or from multiple views by perturbing an m-rep according to an approximation of the orientation, scale and projective geometry of the scene. Such an application can be thought of as a  $3D$ -from- $2D$  deformable model segmentation, open to various semiautomatic approaches. Thus, m-reps would be used as  $3D$  templates for atlas-driven (or interactive) acquisition

of textured geometry data from image data. Local medialness statistics might be derivable from image-based data, using techniques related to the core-atom statistics of Stetten [138], as described in Sec. 2.4.2. Such methods would classify image regions in terms of “tube-like,” “slab-like,” or “sphere-like” geometry with axial directions and radii, allowing m-rep templates to be selected for specific object types.

### **Volume rendering and volume graphics using m-reps**

Chen explored the use of m-reps for segmenting and volume-rendering objects in medical images. These methods were discussed in Sec. 2.4.1. Current m-reps are even more suited for use as deformable volume-primitives for volume rendering. They can be used, as per Gagvani, for the segmentation and subsequent deformation and animation of objects in volume data; or they can be used directly as deformable, multiscale volume primitives for volume graphics applications.

### **Multiscale, physically based properties of m-reps**

Because m-reps are fundamentally a deformable volume representation, they are well-suited to computing physical properties for solid models under deformation. Physically based modeling with m-reps has been explored already in the FEM work of Crouch, as discussed in Sec. 6.3.3. An especially exciting aspect of this work was the use of the m-rep-based multiscale refinement scheme in the FEM equation solver. Physically based applications can expect to draw on the multiscale nature of m-reps to provide approximate and on-the-fly adaptive computation of physically based properties of models under deformation. Sec. 2.5.5 discussed some of the work by Peters and Nasri [116] on defining inertial moments and other properties of subdivision surface bounded models; the work of Stetten [139] on determining volume for objects based on medial descriptions is also relevant.

M-reps would easily provide an LOD-adaptive refinement approach to finding inertial moments under deformation, using both the figural structure of the model and the varying subdivision at the boundary to compute the physical properties. They could use approximations based on medial structure or coarse boundary involute mesh, then compute for subdivision boundaries, then for displaced surfaces. Such methods may assume uniform density for the object, or impose density or composition functions on the m-rep solid based on the  $(u, v, t, \tau)$  for  $\tau < 1$ , though behavior at join regions would need to be defined. Such functions would share invariance under

similarity transformation as well as the object-based deformability of the coordinate system, allowing them to be “carried along” as the object is deformed.

More possibilities arise from the FEM work of Crouch. In finite-element analysis, the linear elasticity equation  $K\underline{a} = \underline{f}$  is solved for  $\underline{a}$ , the displacements at the mesh nodes. Diagonalizing the stiffness matrix  $K$  (or, rather, a reduced  $K_R$ , since  $K$  is provably singular, see Crouch [33]) produces an orthogonal coordinate system whose scaled axes describe the eigenmodes of the deformation, in terms of which solving for  $\underline{a}$  is extremely efficient. If corresponding eigenmodes of mechanical deformation in terms of non-linear Lie group geometric deformations of m-reps can be derived, then finite-element solutions from these m-reps could be computed with similar efficiency.

Collision detection is another area where m-reps might find use as deformable models with both a figural, solid-object aspect and an LOD-based boundary description provided by their subdivision surfaces. Because stationary subdivision surfaces have the convex-hull property, the inverted generating meshes for the interpolating boundaries can be used as coarse approximations to the object shape, and the figure-subfigure decomposition could be used as a coarse model description. Better  $(x, y, z)$ -to-subdivision boundary distance functions would have to be implemented; the strictly local nature of the current  $(x, y, z)$ -to- $(u, v)$  nearpoint code is unacceptable when robust measures of correct geometric distance are required. This remains an open problem in the subdivision surface community, though numerous heuristics might be applied to produce adequate metrics for specific applications. A true distance-at-scale function, as discussed in Sec. 6.2.2, might also be worth exploring for use in a multiscale collision detection algorithm based on iterative refinement of a fuzzy distance function.

### 7.3.3 Final remarks on future m-rep applications

It is almost certain that m-reps will continue to be used extensively in medical image research in MIDAG and increasingly by other research labs as well. In such future research, they will benefit from many of the conceived changes and new modeling paradigms discussed above.

Whether m-reps will be developed as tolerance-based multifigure primitives for general CAD and modeling applications remains to be seen. Whatever methods are eventually developed in these fields must address the same issues that m-reps do: the separation of object-based deformability vs. fine detail, the need for figural shape descriptions and deformability, the representational transparency and polymorphism

issues, etc. Computer graphics and CAGD stand to benefit from new geometric primitives that can serve as robust, statistical shape descriptors based on scale space ideas and ideology. This, however, must await future research.

## 7.4 Personal words

In the course of this work, many insights were gained into the problems of modeling with tolerance-based medial primitives, insights that have pointed to new possibilities for modeling methodologies for CAGD and shape-analysis. While m-rep research is still young, it has already shown the value and power of tolerance-based medial models as geometric representations, particularly in  $3D$  medical imaging applications.

When I first conceived, with Steve Pizer and Turner Whitted and Dave Chen, that *cores*—multiscale medial axes—might be used as new graphics and modeling primitives, I couldn't have imagined the work that would lead to discrete m-reps: the efforts of dozens of researchers, the conference papers and journal articles and theses, the research grants and patents and incorporations. In my years as a graduate student, I saw and participated in the transition of multiscale medial ideas from elegant—albeit incomplete—theories and research prototypes to clinical applications and profound mathematical insights. From the seed of an idea—that all measurements are made with a finite aperture, a spiritual descendent of Heisenberg's Principle—grew a redefinition of the way we define *shape*, and this new understanding has been applied to medical technology with the potential to save lives and improve the human condition. I feel both proud and humble to have had a part in this work.

# Bibliography

- [1] Brett Allen, Brian Curless, and Zoran Popović. Articulated body deformation from range scan data. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 612–619. ACM Press, 2002.
- [2] Nina Amenta, Marshall Bern, and Manolis Kamvysselis. A new Voronoi-based surface reconstruction algorithm. *Computer Graphics*, 32(Annual Conference Series):415–421, August 1998.
- [3] D. Attali and A. Montanvert. Computing and simplifying 2D and 3D continuous skeletons. *Computer Vision and Image Understanding*, 67(3):261–273, sept 1997.
- [4] R. Avila, L. Sobierajski, and A. Kaufman. Towards a comprehensive volume visualization system. In *Proceedings of IEEE Visualization '92*, pages 13–20. IEEE Computer Society Press, October 1992.
- [5] Chandrajit L. Bajaj and Insung Ihm. Smoothing polyhedra using implicit algebraic splines. *Computer Graphics*, 26(2):79–88, July 1992.
- [6] Gill Barequet and Micha Sharir. Piecewise-linear interpolation between polygonal slices. *Computer Vision and Image Understanding: CVIU*, 63(2):251–272, March 1996.
- [7] A. H. Barr. Global and local deformations of solid primitives. *ACM Computer Graphics, (Proceedings of SIGGRAPH 84)*, 18(3):21–30, July 1984.
- [8] Henning Biermann, Daniel Kristjansson, and Denis Zorin. Approximate Boolean operations on free-form solids. In *SIGGRAPH 2001 Conference Proceedings*, Annual Conference Series, pages 185–194. ACM SIGGRAPH, Addison Wesley, August 2001.
- [9] Henning Biermann, Adi Levin, and Denis Zorin. Piecewise smooth subdivision surfaces with normal control. In *SIGGRAPH 2000, Computer Graphics Proceedings*, Annual Conference Series, pages 113–120. ACM Press, Addison Wesley Longman, 2000.

- [10] Robert Blanding, Cole Brooking, Mark Ganter, and Duane Storti. A skeletal-based solid editor. In Willem F. Bronsvort and David C. Anderson, editors, *Proceedings of the Fifth Symposium on Solid Modeling and Applications (SSMA-99)*, pages 141–150, New York, June 9–11 1999. ACM Press.
- [11] James F. Blinn. A generalization of algebraic surface drawing. *ACM Transactions on Graphics*, 1(3):235–256, July 1982.
- [12] James F. Blinn and Martin E. Newell. Texture and reflection in computer generated images. *Communications of the ACM*, 19(10):542–547, 1976.
- [13] Jules Bloomenthal. Medial-based vertex deformation. In *Proceedings of the 2002 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 147–151. ACM Press, 2002.
- [14] Jules Bloomenthal and Ken Shoemake. Convolution surfaces. *Computer Graphics (SIGGRAPH '91 Proceedings)*, 25(4):251–256, July 1991.
- [15] Jules Bloomenthal and Brian Wyvill. Interactive techniques for implicit modeling. *Computer Graphics (1990 Symposium on Interactive 3D Graphics)*, 24(2):109–116, March 1990.
- [16] H. Blum. A transformation for extracting new descriptors of shape. In W. Wathen-Dunn, editor, *Models for the Perception of Speech and Visual Forms*, pages 362–380. MIT Press, Amsterdam, 1967.
- [17] H. Blum. Biological shape and visual science (Part I). *J. Theoretical Biology*, 38:205–287, 1973.
- [18] H. Blum and R. Nagel. Shape description using weighted symmetric axis features. *Pattern Recognition*, 10, 1978.
- [19] Jonathan W. Brandt. Describing a solid with the three-dimensional skeleton. In Joseph D. Warren, editor, *Proceedings of the International Society for Optical Engineering: Curves and Surfaces in Computer Vision and Graphics III*, volume 1830, pages 258–269, Boston, MA, November 16–18 1992. SPIE.
- [20] C. A. Burbeck, S. M. Pizer, B. S. Morse, D. Ariely, G. S. Zauberman, and J. Rolland. Linking object boundaries at scale: a common mechanism for size and shape judgements. Technical Report TR94-041, Department of Computer Science, University of North Carolina - Chapel Hill, 1994.
- [21] Christina A. Burbeck and Stephen M. Pizer. Object representation by cores: Identifying and representing primitive spatial regions. Technical Report TR94-048b, Department of Computer Science, University of North Carolina - Chapel Hill, July 27 1994.

- [22] N. Burtnyk and M. Wein. Interactive skeleton techniques for enhancing motion dynamics in key frame animation. *Communications of the ACM*, 19(10):564–569, October 1976.
- [23] B. M. Cameron and R. A. Robb. An axial skeleton based surface deformation algorithm for patient specific anatomic modeling. In J. D. Westwood, editor, *Medicine Meets Virtual Reality 2000 Proceedings*, pages 53–58. IOS Press, 2000.
- [24] Steve Capell, Seth Green, Brian Curless, Tom Duchamp, and Zoran Popović. Interactive skeleton-driven dynamic deformations. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 586–593. ACM Press, 2002.
- [25] Jonathan C. Carr, Tim J. Mitchell, Richard K. Beatson, Jon B. Cherrie, W. Richard Fright, Bruce C. McCallum, and Tim R. Evans. Reconstruction and representation of 3D objects with radial basis functions. In Stephen Spencer, editor, *Proceedings of the Annual Computer Graphics Conference (SIGGRAPH-01)*, pages 67–76, New York, August 12–17 2001. ACM Press.
- [26] E. Catmull and J. Clark. Recursively generated B-spline surfaces on arbitrary topological meshes. *Computer-Aided Design*, 10:350–355, September 1978.
- [27] Edward E. Catmull. *A Subdivision Algorithm for Computer Display of Curved Surfaces*. PhD thesis, Computer Science Department, University of Utah, December 1974. Report UTEC-CSs-74-133.
- [28] David Chen, Stephen M. Pizer, and J. Turner Whitted. Using multiscale medial models to guide volume visualization. Technical Report TR99-014, University of North Carolina, Chapel Hill, March 2, 1999.
- [29] David Taiwie Chen. *Volume Rendering Guided by Multiscale Medial Models*. Ph. D. thesis, University of North Carolina at Chapel Hill, 1997. Department of Computer Science.
- [30] Greg Clary. *Image Sequence Classification via Anchor Primitives*. Ph. D. thesis, University of North Carolina at Chapel Hill, 2003. Department of Computer Science.
- [31] Jonathan Cohen, Amitabh Varshney, Dinesh Manocha, Greg Turk, Hans Weber, Pankaj Agarwal, Frederick Brooks, and William Wright. Simplification envelopes. *Computer Graphics*, 30(Annual Conference Series):119–128, 1996.
- [32] Robert L. Cook. Shade trees. In *Computer Graphics (Proceedings of SIGGRAPH 84)*, volume 18, pages 223–231, July 1984.
- [33] Jessica Crouch. *Medial Techniques for Automating Finite Element Analysis*. Ph. D. thesis, University of North Carolina at Chapel Hill, August 2003. Department of Computer Science.

- [34] Jessica R. Crouch. In conversation. Summer, 2003.
- [35] Jessica R. Crouch, Stephen M. Pizer, Edward R. Chaney, and Marco Zaider. Medially based meshing with finite element analysis of prostate deformation. In *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2003 6th International Conference, Montréal, Canada, November 15-18, 2003, Proceedings, Part I*, pages 108–115. Springer-Verlag, 2003.
- [36] Tim Culver. *Computing the Medial Axis of a Polyhedron Reliably and Efficiently*. Ph. D. thesis, Department of Computer Science, University of North Carolina at Chapel Hill, 2000.
- [37] James N. Damon. Properties of ridges and cores for two-dimensional images. Technical report, Department of Mathematics, University of North Carolina at Chapel Hill, 1995.
- [38] James N. Damon. Smoothness and geometry of boundaries associated to skeletal structures. Technical report, Department of Mathematics, University of North Carolina at Chapel Hill, 2001.
- [39] James N. Damon. Determining the geometry of boundaries of objects from medial data. Technical report, Department of Mathematics, University of North Carolina at Chapel Hill, 2002.
- [40] James N. Damon. Smoothness and geometry of boundaries associated to skeletal structures I: sufficient conditions for smoothness. Technical report, Department of Mathematics, University of North Carolina at Chapel Hill, 2002.
- [41] James N. Damon. Smoothness and geometry of boundaries associated to skeletal structures II: geometry in the Blum case. Technical report, Department of Mathematics, University of North Carolina at Chapel Hill, 2002.
- [42] Rhodri H. Davies, Carole J. Twining, Tim F. Cootes, John C. Waterton, and Chris J. Taylor. 3D statistical shape models using direct optimization of description length. In *Proceedings of ECCV'02, Vol. III*, pages 3–20, 2002.
- [43] Tony DeRose, Michael Kass, and Tien Truong. Subdivision surfaces in character animation. *Computer Graphics*, 32(Annual Conference Series):85–94, August 1998.
- [44] Mathieu Desbrun, Mark Meyer, Peter Schröder, and Alan H. Barr. Discrete differential-geometry operators in  $nD$ . Technical report (submitted for publication), Caltech Multi-Res Modeling Group, July 2000.
- [45] Pavel Dimitrov, James N. Damon, and Kaleem Siddiqi. Flux invariants for shape. In *Proceedings of CVPR'03, Vol. I*, pages 835–841, Madison, WI, June 16–22 2003.

- [46] Michael Doggett. ATI Research. *Displacement Mapping*, From a tutorial session at GDC 2003, January 2003.
- [47] D. Doo and M. Sabin. Behaviour of recursive division surfaces near extraordinary points. *Computer-Aided Design*, 10:356–360, 1978.
- [48] Nira Dyn, David Levin, and John A. Gregory. A butterfly subdivision scheme for surface interpolation with tension control. *ACM Transactions on Graphics*, 9(2):160–169, April 1990.
- [49] David Eberly. *Geometric Methods for Analysis of Ridges in N-Dimensional Images*. Ph. D. thesis, Department of Computer Science, University of North Carolina at Chapel Hill, 1994.
- [50] Gerald Farin. *Curves and Surfaces for Computer-Aided Geometric Design*. Academic Press, New York, NY, USA, fourth edition, 1997.
- [51] R. T. Farouki and C. A. Neff. Hermite interpolation by Pythagorean hodograph quintics. *Mathematics of Computation*, 64(212):1589–1609, October 1995.
- [52] R. T. Farouki and T. Sakkalis. Pythagorean hodographs. *IBM Journal of Research and Development*, 34(5):736–752, September 1990.
- [53] P. Thomas Fletcher. Blending and displaying multigure m-reps using implicit surfaces. Comp236 Course project, Spring semester, Department of Computer Science, University of North Carolina - Chapel Hill, 1999.
- [54] P. Thomas Fletcher. A software project fitting bezier splines to a sampled medial quad-mesh. Unpublished software implementation, Department of Computer Science, University of North Carolina - Chapel Hill, 1999.
- [55] P. Thomas Fletcher, Sarang Joshi, Conglin Lu, and Stephen Pizer. Gaussian distributions on lie groups and their application to statistical shape analysis. In *Information Processing in Medical Imaging IPMI '03, Lecture Notes in Computer Science*, 2003.
- [56] P. Thomas Fletcher, Conglin Lu, and Sarang Joshi. Statistics of shape via principal component analysis on lie groups. In *Proceedings of CVPR 2003*, 2003.
- [57] P. Thomas Fletcher, Stephen M. Pizer, Andrew Thall, and A. Graham Gash. Shape modeling and image visualization in 3D with m-rep object models. Technical report, Dept. of Computer Science, UNC Chapel Hill, 2000.
- [58] P. Thomas Fletcher, Andrew Thall, Daniel S. Fritsch, and Yoni Fridman. SCAMP: A solid modeling program using slice-constrained medial primitives for modeling 3D anatomical objects. technical report TR99-035, Department of Computer Science, University of North Carolina - Chapel Hill, December 06 1999.

- [59] James D. Foley, Andries van Dam, Steven K. Feiner, and John F. Hughes. *Computer Graphics: Principles and Practice, Second Edition in C*. Addison-Wesley Publishing Company, 2nd edition, 1996.
- [60] Sarah F. Frisken, Ronald N. Perry, Alyn P. Rockwood, and Thouis R. Jones. Adaptively sampled distance fields: A general representation of shape for computer graphics. In Kurt Akeley, editor, *Siggraph 2000, Computer Graphics Proceedings*, Annual Conference Series, pages 249–254. ACM Press, Addison Wesley Longman, 2000.
- [61] D. Fritsch, E. Chaney, A. Boxwala, M. McAuliffe, S. Raghavan, A. Thall, and J. Earnhart. Core-based portal image registration for automatic radiotherapy treatment verification. *International Journal of Radiation, Oncology, Biology, Physics; special issue on Conformal Therapy*, 5(33):1287–1300, 1995.
- [62] Daniel S. Fritsch. *Registration of Radiotherapy Images Using Multiscale Medial Descriptions of Image Structure*. Ph. D. thesis, Department of Computer Science, University of North Carolina at Chapel Hill, 1993.
- [63] Daniel S Fritsch, Stephen M Pizer, Liyun Yu, Valen Johnson, and Edward L Chaney. Localization and segmentation of medical image objects using deformable shape loci. In James Duncan and Gene Gindi, editors, *Information Processing in Medical Imaging IPMI '97, Lecture Notes in Computer Science*, volume 1230, pages 127–140. Springer-Verlag, 1997. Available as tech report TR96-033 at <ftp://ftp.cs.unc.edu/pub/publications/techreports/96-033.ps.Z>.
- [64] H. Fuchs, Z. Kedmen, and S. Uselton. Optimal surface reconstruction from planar contours. *Communications of the ACM*, 20(10):693–702, October 1977.
- [65] Nikhil Gagvani, Dilip Kenchammana-Hosekote, and Deborah Silver. Volume animation using the skeleton tree (color plate p. 166). In *Proceedings of the 1998 Symposium on Volume Visualization (VOLVIS-98)*, pages 47–54, New York, October 19–20 1998. ACM Press.
- [66] Marie-Paule Gascuel. An implicit formulation for precise contact modeling between flexible solids. In James T. Kajiya, editor, *Computer Graphics (SIGGRAPH '93 Proceedings)*, volume 27, pages 313–320, August 1993.
- [67] Marie-Paule Gascuel, Anne Verroust, and Claude Puech. Animation and collisions between complex deformable bodies. In *Proceedings of Graphics Interface '91*, pages 263–270, June 1991.
- [68] Guido Gerig, Keith E. Muller, Emily O. Kistner, Yueh-Yun Chi, Miranda Chakos, Martin Styner, and Jeffrey A. Lieberman. Age and treatment related local hippocampal changes in schizophrenia explained by a novel shape analysis method. In *Medical Image Computing and Computer-Assisted Intervention -*

*MICCAI 2003 6th International Conference, Montréal, Canada, November 15-18, 2003, Proceedings, Part II*, pages 653–660. Springer-Verlag, 2003.

- [69] P. Giblin and B. Kimia. On the intrinsic reconstruction of shape from its symmetries. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 25(7):895–911, July 2003.
- [70] Peter Giblin. Symmetry sets and medial axes in two and three dimensions. In Roberto Cipolla and Ralph Martin, editors, *Proceedings of the 9th IMA Conference on the Mathematics of Surfaces (IMA-00)*, volume IX of *The Mathematics of Surfaces*, pages 306–321, London, Berlin, Heidelberg, September 4–7 2000. Springer.
- [71] Gene H. Golub and Charles F. Van Loan. *Matrix Computation*. Series in the Mathematical Sciences. Johns Hopkins University Press, Baltimore and London, 3rd edition, 1996.
- [72] Carlos Gonzalez-Ochoa, Scott McCammon, and Jörg Peters. Computing moments of objects enclosed by piecewise polynomial surfaces. *ACM Transactions on Graphics*, 17(3):143–157, July 1998.
- [73] Eitan Grinspun and Peter Schröder. Normal bounds for subdivision-surface interference detection. Submitted for publication, 2001.
- [74] S. Grossberg and E. Mingolla. Neural dynamics of surface perception: Boundary webs, illuminants, and shape-for-shading. *Computer Vision, Graphics, and Image Processing*, 37:116–165, 1987.
- [75] Robert Grzeszczuk, Chris Henn, and Roni Yagel. *Advanced Geometric Techniques for Ray Casting Volumes*. SIGGRAPH 98 Course Notes #4, ACM, July 1998.
- [76] Igor Guskov, Kiril Vidimčec, Wim Sweldens, and Peter Schröder. Normal meshes. In Kurt Akeley, editor, *Siggraph 2000, Computer Graphics Proceedings*, Annual Conference Series, pages 95–102. ACM Press, Addison Wesley Longman, 2000.
- [77] Mark Halstead, Michael Kass, and Tony DeRose. Efficient, fair interpolation using Catmull-Clark surfaces. In James T. Kajiya, editor, *Computer Graphics (SIGGRAPH '93 Proceedings)*, volume 27, pages 35–44, August 1993.
- [78] Qiong Han, Conglin Lu, Guodong Liu, Stephen M. Pizer, Sarang Joshi, and Andrew Thall. Representing multi-figure anatomical objects. In *Proceedings of the IEEE International Symposium on Biomedical Imaging (ISBI 2004), Arlington, VA, April 15-18, To appear.*, 2004.

- [79] Paul S. Heckbert. Ray tracing JELL-O brand gelatin. In *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, pages 73–74. ACM Press, 1987.
- [80] Gentaro Hirota, Renee Maheshwari, and Ming C. Lin. Fast volume-preserving free form deformation using multi-level optimization. In Willem F. Bronsvort and David C. Anderson, editors, *Proceedings of the Fifth Symposium on Solid Modeling and Applications (SSMA-99)*, pages 234–245, New York, June 9–11 1999. ACM Press.
- [81] Kenneth E. Hoff III, John Keyser, Ming Lin, Dinesh Manocha, and Tim Culver. Fast computation of generalized Voronoi diagrams using graphics hardware. *Computer Graphics*, 33(Annual Conference Series):277–286, 1999.
- [82] Hugues Hoppe, Tony DeRose, Tom Duchamp, Mark Halstead, Hubert Jin, John McDonald, Jean Schweitzer, and Werner Stuetzle. Piecewise smooth surface reconstruction. *Computer Graphics*, 28(Annual Conference Series):295–302, July 1994.
- [83] R. A. Hummel. Representations based on zero crossings in scale space. In *Proceedings, CVPR '86 (IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Miami Beach, FL, June 22–26, 1986)*, IEEE Publ.86CH2290-5, pages 204–209. IEEE, 1986.
- [84] Takeo Igarashi, Satoshi Matsuoka, and Hidehiko Tanaka. Teddy: A sketching interface for 3d freeform design. *Proceedings of SIGGRAPH 99*, pages 409–416, August 1999. ISBN 0-20148-560-5. Held in Los Angeles, California.
- [85] David J. Jackson. Boundary representation modelling with local tolerances. In *SMA '95: Proceedings of the Third Symposium on Solid Modeling and Applications*, pages 247–254. ACM, May 1995. held May 17-19, 1995 in Salt Lake City, Utah.
- [86] Sarang Joshi, Stephen Pizer, P. Thomas Fletcher, Paul Yushkevich, Andrew Thall, and J. S. Marron. Multiscale deformable model segmentation and statistical shape analysis using medial descriptions. *IEEE Trans. on Medical Imaging*, 21(5):538–550, May 2002.
- [87] Sarang Joshi, Stephen M. Pizer, P. Thomas Fletcher, Andrew Thall, and Greg Tracton. Multi-scale 3D deformable model segmentation based on medial description. In *International Conference on Information Processing in Medical Imaging (IPMI'01)*, pages 64–77, 2001.
- [88] Tao Ju, Frank Losasso, Scott Schaefer, and Joe Warren. Dual contouring of hermite data. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 339–346. ACM Press, 2002.

- [89] Robert A. Katz and Stephen M. Pizer. Untangling the Blum medial axis transform. Technical Report (under review for publication), Department of Computer Science, University of North Carolina at Chapel Hill, 2001.
- [90] Ari Kaufmann. Efficient algorithms for 3D scan-conversion of parametric curves, surfaces and volumes. *Computer Graphics (Proceedings of Siggraph '87)*, 21(4):171–179, July 1987.
- [91] Michael Kerckhove. Unpublished work on interpolation based on a discrete medial sampling. 2003.
- [92] Leif P. Kobbelt, Thilo Bareuther, and Hans-Peter Seidel. Multiresolution shape deformations for meshes with dynamic vertex connectivity. *Computer Graphics Forum*, 19(3), 2000.
- [93] Leif P. Kobbelt, Mario Botsch, Ulrich Schwanecke, and Hans-Peter Seidel. Feature-sensitive surface extraction from volume data. In Eugene Fiume, editor, *SIGGRAPH 2001, Computer Graphics Proceedings, Annual Conference Series*, pages 57–66. ACM Press / ACM SIGGRAPH, 2001.
- [94] J. J. Koenderink. The structure of images. *Biological Cybernetics*, 50:363–370, 1984.
- [95] F. Lazarus, S. Coquillart, and P. Jancene. Axial deformation: An intuitive deformation technique. *Computer Aided Design*, 26(8):607–613, August 1994.
- [96] Aaron Lee, Henry Moreton, and Hugues Hoppe. Displaced subdivision surfaces. In Kurt Akeley, editor, *Siggraph 2000, Computer Graphics Proceedings, Annual Conference Series*, pages 85–94. ACM Press, Addison Wesley Longman, 2000.
- [97] N. Litke, A. Levin, and P. Schroeder. Fitting subdivision surfaces. In *IEEE Visualization 2001*, pages 319–324, October 2001.
- [98] N. Litke, A. Levin, and P. Schroeder. Trimming for subdivision surfaces. *Computer Aided Geometric Design*, 18(5):463–481, 2001.
- [99] Charles T. Loop. Smooth subdivision surfaces based on triangles. Master’s Thesis, University of Utah, August 1987. Department of Mathematics.
- [100] Ron MacCracken and Kenneth I. Joy. Free-form deformations with lattices of arbitrary topology. *Proceedings of SIGGRAPH 96*, pages 181–188, August 1996. ISBN 0-201-94800-1. Held in New Orleans, Louisiana.
- [101] Dinesh Manocha. In conversation, Jan 2004.
- [102] Lee Markosian, Jonathan M. Cohen, Thomas Crulli, and John Hughes. Skin: a constructive approach to modeling free-form shapes. *Computer Graphics, 33(Annual Conference Series)*:393–400, 1999.

- [103] Kevin T. McDonnell, Hong Qin, and Robert A. Wlodarczyk. Virtual clay: A real-time sculpting system with haptic toolkits. *2001 ACM Symposium on Interactive 3D Graphics*, pages 179–190, March 2001. ISBN 1-58113-292-1.
- [104] Henry P. Moreton and Carlo H. Séquin. Functional optimization for fair surface design. *Computer Graphics*, 26(2):167–176, 1992.
- [105] Bryan S. Morse. *Computation of Object Cores from Grey-level Images*. Ph. D. thesis, Department of Computer Science, University of North Carolina at Chapel Hill, 1995.
- [106] L. R. Nackman. Curvature relation in three-dimensional symmetric axes. *Computer Graphics and Image Processing*, 20(1):43–57, September 1982.
- [107] L. R. Nackman. Three-Dimensional Shape Description Using the Symetric Axis Transform. Ph. D. Thesis, University of North Carolina at Chapel Hill, Dept. of Computer Science, 1982.
- [108] L. R. Nackman and S. M. Pizer. Three-dimensional shape description using the symmetric axis transform I:theory. *IEEE Trans. Pattern Analysis and Machine Intelligence*, PAMI-7(2):187–202, 1985.
- [109] H. Nishimura, M. Hirai, T. Kawai, T. Kawata, I. Shirakawa, and K. Omura. Object modeling by distribution function and a method of image generation. In *Trans. IECE Japan, Part D*, volume J68-D(4) of *Proceedings of Electronics Communication Conference '85*, pages 718–725, 1985.
- [110] Byong Mok Oh, Max Chen, Julie Dorsey, and Frdo Durand. Image-based modeling and photo editing. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 433–442. ACM Press, 2001.
- [111] Manuel M. Oliveira, Gary Bishop, and David McAllister. Relief texture mapping. *Proceedings of SIGGRAPH 2000*, pages 359–368, July 2000.
- [112] Ken Perlin. Improving noise. In *Proceedings of the 29th annual conference on computer graphics and interactive techniques*, pages 681–682. ACM Press, 2002.
- [113] P. Perona and J. Malik. Scale space and edge detection using anisotropic diffusion. In *Workshop on Computer Vision, (Miami Beach, FL, November 30 – December 2, 1987)*, pages 16–22, Washington, DC., 1987. IEEE Computer Society Press.
- [114] Ronald N. Perry and Sarah F. Frisken. Kizamu: A system for sculpting digital characters. In Eugene Fiume, editor, *SIGGRAPH 2001, Computer Graphics Proceedings*, Annual Conference Series, pages 47–56. ACM Press / ACM SIGGRAPH, 2001.

- [115] Jörg Peters. Local cubic and bicubic  $C^1$  surface interpolation with linearly varying boundary normal. *Computer Aided Geometric Design*, 7(6):499–516, November 1990.
- [116] Jörg Peters and Ahmad Nasri. Computing volumes of solids enclosed by recursive subdivision surfaces. *Computer Graphics Forum*, 16(3):C89–C94, 1997.
- [117] Jörg Peters and Georg Umlauf. Gaussian and mean curvature of subdivision surfaces. In Roberto Cipolla and Ralph Martin, editors, *Proceedings of the 9th IMA Conference on the Mathematics of Surfaces (IMA-00)*, volume IX of *The Mathematics of Surfaces*, pages 59–69, London, Berlin, Heidelberg, September 4–7 2000. Springer.
- [118] Jörg Peters and Georg Umlauf. Computing curvature bounds for bounded subdivision surfaces. *CAGD: Special Issue on Subdivision*, 18(5):455–461, 2001.
- [119] Roland Pilgram, P. Thomas Fletcher, Stephen M. Pizer, Otmar Pachinger, and Rainer Schubert. Common shape model and inter-individual variations of the heart using medial representation. Technical report, Institute for Medical Knowledge Representation and Visualization, University for Health Information and Technology Tyrol, Austria, February 2004.
- [120] S. Pizer, D. Fritsch, P. Yushkevich, V. Johnson, and E. Chaney. Segmentation, registration, and measurement of shape variation via image object shape. *IEEE Transactions on Medical Imaging*, 18:851–865, October 1999.
- [121] Stephen M. Pizer, James Z. Chen, P. Thomas Fletcher, Yonatan Fridman, Daniel S. Fritsch, A. Graham Gash, John M. Glotzer, Michael R. Jiroutek, Sarang Joshi, Conglin Lu, Keith E. Muller, Andrew Thall, Gregg Tracton, Paul Yushkevich, and Edward L. Chaney. Deformable m-reps for 3D medical image segmentation. *Int. J. of Computer Vision*, 55(3), November 2003.
- [122] Stephen M. Pizer, David Eberly, Daniel S. Fritsch, and Bryan S. Morse. Zoom-invariant vision of figural shape: The mathematics of cores. *Computer Vision and Image Understanding: CVIU*, 69(1):055–071, January 1998.
- [123] Stephen M. Pizer, P. Thomas Fletcher, Andrew Thall, Martin Styner, Guido Gerig, and Sarang Joshi. Object models in multiscale intrinsic coordinates via m-reps. Technical report, Dept. of Computer Science, University of North Carolina at Chapel Hill. Presented at the First International Workshop on Generative Model-Based Vision (GMBV 2002), Copenhagen, Denmark, June, 2002.
- [124] Plato. *The Republic*, volume VII. Athens, Greece, 360 B.C.
- [125] William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical Recipes in C, The Art of Scientific Computing*. Cambridge University Press, 2nd edition, 1992.

- [126] Manjari I. Rao. Analysis of a Locally Varying Intensity Template for Segmentation of Kidneys in CT Images. Master's Thesis, University of North Carolina at Chapel Hill, 2003. Department of Biomedical Engineering.
- [127] John Rhoades. *Shaping Curved Surfaces*. Ph. D. thesis, University of North Carolina at Chapel Hill, 1994. Department of Computer Science.
- [128] Azriel Rosenfeld. Axial representations of shape. *Computer Vision, Graphics, and Image Processing*, 33(2):156–173, February 1986.
- [129] Stan Sclaroff and Alex Pentland. Generalized implicit functions for computer graphics. *Computer Graphics (SIGGRAPH '91 Proceedings)*, 25(4):247–250, July 1991.
- [130] T. W. Sederberg and S. R. Parry. Free-form deformation of solid geometric models. *Computer Graphics (SIGGRAPH '86 Proceedings)*, 20(4):151–160, August 1986.
- [131] M. Shapira and A. Rappoport. Shape blending using the star-skeleton representation. *IEEE Computer Graphics & Applications*, 15(2):44–50, March 1995.
- [132] Andrei Sherstyuk. Ray-tracing implicit surfaces: a generalized approach. Technical Report 96/290, Dept. Computer Science, Monash University, Australia 3168, December 1996.
- [133] Andrei Sherstyuk. *Convolution surfaces in computer graphics*. Ph. D. thesis, Department of Computer Science, Monash University, Australia, 1999.
- [134] Andrei Sherstyuk. Shape design using convolution surfaces. In *Proceedings of Shape Modeling International '99*, March 1999.
- [135] Ken Shoemake. Animating rotation with quaternion curves. *Computer Graphics*, 19(3):245–254, July 1985.
- [136] K. Siddiqi, S. Bouix, A. Tannenbaum, and S. Zucker. The hamilton-jacobi skeleton. In *Proceedings of the 7th IEEE International Conference on Computer Vision (ICCV-99)*, volume II, pages 828–834, Los Alamitos, CA, September 20–27 1999. IEEE.
- [137] Jos Stam. Exact evaluation of catmull-clark subdivision surfaces at arbitrary parameter values. In Michael Cohen, editor, *SIGGRAPH 98 Conference Proceedings*, Annual Conference Series, pages 395–404. ACM SIGGRAPH, Addison Wesley, July 1998.
- [138] G. Stetten, R. Landesman, and S. Pizer. Core-atoms and the spectra of scale. In *Proceedings of SPIE Medical Imaging '97 Conference, part 2*, volume 3034, pages 642–652, 1997.

- [139] George Stetten. Volume of arbitrary shapes from boundary curvature and medial scale. Technical Report TR99-001, Department of Computer Science, University of North Carolina - Chapel Hill, January 04 1999.
- [140] George D. Stetten and Stephen M. Pizer. Medial node models to identify and measure objects in real-time 3d echocardiography. *IEEE Transactions on Medical Imaging*, 18(10):1025–1034, 1999.
- [141] Eric J Stollnitz, Tony D. DeRose, and David H. Salesin. *Wavelets for Computer Graphics: Theory and Applications*. Morgan Kaufmann, San Francisco, CA, USA, first edition, 1996.
- [142] Duane W. Storti, George M. Turkiyyah, Mark A. Ganter, Chek T. Lim, and Derek M. Stat. Skeleton-based modeling operations on solids. In *SMA '97: Proceedings of the Fourth Symposium on Solid Modeling and Applications*, pages 141–154. ACM, May 1997. held May 14-16, 1997 in Atlanta, Georgia.
- [143] Gilbert Strang. *Introduction to Applied Mathematics*. Wellesley-Cambridge Press, Cambridge, MA, 1986.
- [144] Martin Styner and Guido Gerig. Three-dimensional medial shape representation incorporating object variability. In *Proceedings of CVPR*, pages 651–656. IEEE Computer Society, December 2001.
- [145] Martin Styner, Guido Gerig, and Jeffrey A. Lieberman. Boundary and medial shape analysis of the hippocampus in schizophrenia. In *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2003 6th International Conference, Montréal, Canada, November 15-18, 2003, Proceedings, Part II*, pages 464–471. Springer-Verlag, 2003.
- [146] Martin A. Styner. *Combined Boundary-Medial Shape Description of Variable Biological Shapes*. Ph. D. thesis, Department of Computer Science, University of North Carolina at Chapel Hill, 2001.
- [147] Marek Teichmann and Seth Teller. Assisted articulation of closed polygonal models. In B. Arnaldi and G. Hegron, editors, *Computer Animation and Simulation '98*, SpringerComputerScience, pages 87–101. Springer-Verlag Wien New York, 1998. Proceedings of the Eurographics Workshop in Lisbon, Portugal, August 31–September 1, 1998.
- [148] Andrew Thall, P. Thomas Fletcher, and Stephen M. Pizer. Deformable solid modeling using sampled medial surfaces: a multiscale approach. Technical report TR00-005, University of North Carolina, Chapel Hill, 2000.
- [149] Greg Turk. Generating textures for arbitrary surfaces using reaction-diffusion. In Thomas W. Sederberg, editor, *Computer Graphics (SIGGRAPH '91 Proceedings)*, volume 25, pages 289–298, July 1991.

- [150] Greg Turk. Texture synthesis on surfaces. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 347–354. ACM Press, 2001.
- [151] C. W. A. M. van Overveld and Brian Wyvill. Phong normal interpolation revisited. *ACM Transactions on Graphics*, 16(4):397–419, October 1997. ISSN 0730-0301.
- [152] Gokul Varadhan, Shankar Krishnan, Young J. Kim, Suhas Diggavi, and Dinesh Manocha. Efficient max-norm distance computation and reliable voxelization. In *Proceedings of the Eurographics/ACM SIGGRAPH symposium on Geometry processing*, pages 116–126. Eurographics Association, 2003.
- [153] Y. Sampath K. Vetsa, Martin Styner, Stephen M. Pizer, Jeffrey A. Lieberman, and Guido Gerig. Caudate shape discrimination in schizophrenia using template-free non-parametric tests. In *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2003 6th International Conference, Montréal, Canada, November 15-18, 2003, Proceedings, Part II*, pages 661–669. Springer-Verlag, 2003.
- [154] Li-Yi Wei and Marc Levoy. Texture synthesis over arbitrary manifold surfaces. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 355–360. ACM Press, 2001.
- [155] J. Weickert, S. Ishikawa, and A. Imiya. On the history of gaussian scale-space axiomatics. In J. Sporring, M. Nielsen, L. Florack, and P. Johansen, editors, *Gaussian Scale-Space Theory, Computational Imaging and Vision*, chapter 4. Kluwer Academic Publishers, 1997.
- [156] Ross T. Whitaker. Geometry-limited diffusion in the characterization of geometric patches in images. *Computer Vision, Graphics, and Image Processing. Image Understanding*, 57(1):111–120, January 1993.
- [157] Ross T. Whitaker. *Geometry-Limited Diffusion*. Ph. D. thesis, Department of Computer Science, University of North Carolina at Chapel Hill, 1994. Wed, 26 Jun 1996 18:14:03 GMT.
- [158] A. Wilson, V. Johnson, S. M. Pizer, D. S. Fritsch, L. Yu, and E. Chaney. Towards a framework for automated image analysis. In *Image Fusion and Shape Variability Techniques, Proc. 16th Leeds Annual Statistical Research Workshop*. University of Leeds Press, 1996.
- [159] Andrew P. Witkin. Scale-space filtering. In Alan Bundy, editor, *Proceedings of the 8th International Joint Conference on Artificial Intelligence*, pages 1019–1022, Karlsruhe, FRG, August 1983. William Kaufmann.

- [160] Geoff Wyvill, Craig McPheeters, and Brian Wyvill. Animating *soft* objects. *The Visual Computer*, 2(4):235–242, 1986.
- [161] Geoff Wyvill, Craig McPheeters, and Brian Wyvill. Data structure for *soft* objects. *The Visual Computer*, 2(4):227–234, 1986.
- [162] Shin Yoshizawa, Alexander G. Belyaev, and Hans-Peter Seidel. Free-form skeleton-driven mesh deformations. In *Proceedings of the eighth ACM symposium on Solid modeling and applications*, pages 247–253. ACM Press, 2003.
- [163] Paul Yushkevich, P. Thomas Fletcher, Sarang Joshi, Andrew Thall, and Stephen M. Pizer. Continuous medial representations for geometric object modeling in 2D and 3D. Technical Report TR02-003, Dept. of Computer Science, University of North Carolina at Chapel Hill. Presented at the First International Workshop on Generative Model Based Vision (GMBV 2002), Copenhagen, Denmark, June, 2002.
- [164] Paul Yushkevich, Stephen M. Pizer, and Tim Culver. Statistical object shape via a medial representation. Technical Report TR00-002, Dept. of Computer Science, UNC Chapel Hill, 1999.
- [165] Paul Yushkevich, Stephen M. Pizer, Sarang Joshi, and J. S. Marron. Intuitive, localized analysis of shape variability. In *International Conference on Information Processing in Medical Imaging (IPMI'01)*, pages 402–408, 2001.
- [166] Denis Zorin, Peter Schröder, and Wim Sweldens. Interpolating subdivision for meshes with arbitrary topology. In Holly Rushmeier, editor, *SIGGRAPH 96 Conference Proceedings*, Annual Conference Series, pages 189–192. ACM SIGGRAPH, Addison Wesley, August 1996. held in New Orleans, Louisiana, 04-09 August 1996.

# Colophon

This dissertation was typeset using the  $\text{\LaTeX}2_{\epsilon}$  typesetting system using MikTeX and WinEdt 5.3. Figures were drawn using `xfig` on a Linux box, and screen-images were captured and exported to `.eps` using PaintShop Pro. Code development on this project used `g++` and Visual C++ 6.0; GUI design used FLTK 1.1.x on both Unix and Windows platforms, as well as Paul Rademacher's GLUI toolkit.