

CONVOLUTION SURFACES IN COMPUTER GRAPHICS

Thesis by
Andrei Sherstyuk

In Fulfillment of the Requirements for the Degree of
Doctor of Philosophy

School Of Computer Science and Software Engineering
Monash University
Australia

Submitted January 17, 1999

©Andrei Sherstyuk 1998

Contents

Title Page	i
Table of Contents	ii
List of Figures	v
List of Tables	vii
Summary	viii
Statement	ix
Acknowledgments	x
1 Introduction	1
1.1 A brief history of implicit surfaces and a literature review	1
1.2 Generalization of classic implicit models	2
1.3 Open questions	3
1.3.1 Formulation	3
1.3.2 Modeling	4
1.3.3 Rendering	4
1.4 Contributions	5
2 Formulation of Convolution Surfaces	7
2.1 Introduction	7
2.2 Definitions	8
2.3 Kernels and primitives	9
2.3.1 A new kernel function	9
2.3.2 The seven kernels	10
2.3.3 Primitives/kernels compatibility chart	11
2.3.4 Computational costs	11
2.3.5 Problems with polynomial kernels	13
2.3.6 A short summary	14
2.4 Development of implicit primitives	14
2.4.1 Point sources	15
2.4.2 Line segments	15
2.4.3 Arcs	15
2.4.4 Triangles	16
2.4.5 Planes	17
2.5 Examples	17
2.6 Performance	18
2.7 Conclusions	19
3 Modeling with Convolution Surfaces	21
3.1 Introduction	21
3.2 Definitions	21
3.2.1 Implicit surfaces	21
3.2.2 Convolution surfaces	21
3.2.3 Implicit primitives	22
3.2.4 Skeletons	22

3.3	The design system	22
3.3.1	New modeling primitives as skeletal elements	22
3.3.2	Local properties of implicit primitives	23
3.3.3	Materials and elements	24
3.3.4	Profiling as a means for achieving variety	25
3.3.5	Offset surfaces as visual aid and the data structures	26
3.3.6	Variables	26
3.4	The main modeling loop	26
3.4.1	Datasets	26
3.4.2	The main modeling strategy	26
3.5	Practical examples of implicit design	27
3.5.1	Global skeleton manipulations	27
3.5.2	Local modeling techniques	30
3.5.3	Volumetric detail	30
3.6	Implementation details and timing results	32
3.7	Conclusion	33
4	Rendering Convolution Surfaces	35
4.1	Introduction	35
4.1.1	The problem	35
4.1.2	Previous work	35
4.1.3	Algorithm preconditions	36
4.1.4	Algorithm postconditions	36
4.2	The algorithm	37
4.2.1	Ray-tracing algorithm for metaballs	37
4.2.2	Generalization of the algorithm for metaballs	37
4.2.3	An example	38
4.2.4	Bounding volumes	39
4.2.5	Shading and texturing	39
4.3	Error analysis	39
4.4	Optimizations	40
4.4.1	Polynomization on demand	40
4.4.2	Fast ray/surface rejection test	40
4.4.3	Volatile and permanent clusters	41
4.4.4	Reusing interpolants	41
4.5	Examples	41
4.6	Conclusions	43
5	Epilogue	47
A	Field functions for point primitives	49
A.1	Cauchy function	49
A.2	Gaussian function	49
A.3	Inverse function	50
A.4	Inverse squared function	50
A.5	Metaballs	50
A.6	Soft objects	50
A.7	W-quartic polynomial	50
B	Field functions for line primitives	51
B.1	Cauchy line segment	51
B.2	Gaussian line segment	51
B.3	Inverse potential line segment	52
B.4	Inverse squared line segment	52
B.5	Polynomial line segment	52

C	RATS Overview and Command Language	53
C.1	Overview	53
C.2	Command Language	54
C.3	Two examples of internal man pages	59
C.3.1	Materials	59
C.3.2	Textures	60
C.4	Selected Datasets	61
	Bibliography	67

List of Figures

1.1	Blinn's Blob.	1
2.1	An implicit surface generated by two point sources.	7
2.2	One-dimensional convolution.	8
2.3	A convolution surface produced by an implicit line segment.	9
2.4	A Gaussian function.	9
2.5	A new kernel function.	10
2.6	Both kernels.	10
2.7	Convolving a line segment with various kernels.	13
2.8	Finding integration domain for line segments.	13
2.9	Finding integration domain for triangles.	14
2.10	Point modeling primitive.	15
2.11	Line modeling primitive.	15
2.12	Arc modeling primitive.	15
2.13	Development of the triangular primitive.	16
2.14	Triangle modeling primitive.	16
2.15	Triangle modeling primitive, revisited.	16
2.16	Family of convolved starfish.	17
2.17	Bounding volumes of the family of convolved starfish.	17
2.18	Implicit sphereflakes.	18
2.19	Blends between primitives of different types.	18
2.20	'Explicit' and 'implicit' coral trees.	18
3.1	Components of a convolution surface.	22
3.2	Implicit primitives.	23
3.3	A skeleton of a coral crab.	23
3.4	A T-shaped convolution surface.	24
3.5	Elements on an implicit icicle.	24
3.6	Individual assignment of local parameters.	25
3.7	Profiling functions.	25
3.8	The main modeling loop.	27
3.9	Shell I. Point-based spherical shell.	28
3.10	Shell II. The offset surface and the convolved surface of an unidentified mollusk.	28
3.11	Shell III. Superposition of two skeletons.	28
3.12	Spindle Cowrie.	29
3.13	Seaweed, modeled with linear profiling functions.	29
3.14	Coral tree, based upon a seaweed model.	29
3.15	Seahorse: A hand drawing, an offset surface and the convolved shape.	29
3.16	Coral crab.	29
3.17	How to make volumetric wrinkles.	30
3.18	Nefertiti, before and after convolution.	31
3.19	Aging Yoda.	31
4.1	Polynomial approximation.	37
4.2	Rendering an implicit icicle.	38
4.3	Implicit cross, front view with uniform error distribution	40
4.4	Implicit cross, side view with visible distortions.	40
4.5	Implicit cross, side view revisited.	40
4.6	Creating permanent clusters.	41
4.7	Implicit sphere-flakes of various complexity.	42

4.8	Time-profiling charts.	42
4.9	Rendering of thin objects and texture mapping.	43
4.10	Coral tree, convolved with a Gaussian kernel.	43
4.11	Hermite crab	43
4.12	“Spinal Starecase”.	44
A.1	Cauchy function.	49
A.2	Gaussian function.	49
A.3	Inverse function.	50
A.4	Inverse squared function.	50
A.5	Metaballs.	50
A.6	Soft objects.	50
A.7	W-quartic polynomial.	50
B.1	Cauchy line segment.	51
B.2	Gaussian line segment.	51
B.3	Inverse potential line segment.	52
B.4	Inverse squared potential line segment.	52
B.5	Polynomial line segment.	52

List of Tables

2.1	Primitives/kernels compatibility chart.	11
2.2	Computational costs for point and line primitives.	12
2.3	Timing test results and speed ratings for points and line segments.	12
2.4	Summary of the timing tests.	12
2.5	Rendering convolved starfish	17
2.6	Timing tests for new modeling functions.	19
3.1	Modeling of spiral seashells.	30
3.2	Summary of modeling techniques.	32
3.3	Rendering time for offset surfaces and convolution surfaces.	32
4.1	Ray-surface equations along the ray's path.	38
4.2	Bounding volumes for modeling primitives.	39
4.3	Optimization methods and rendering times.	41
4.4	Rendering times for the Sphereflake model.	42
4.5	Rendering times for the Spinal Starecase model.	44
4.6	Implemented implicit primitives.	44

Summary

Implicit surfaces obtained by the convolution of mixed dimensional primitives with some potential function, are a generalization of popular implicit surface models: blobs, metaballs and soft objects. These models differ in their choice of potential function but agree upon the use of underlying modeling primitives, namely, points, which puts severe limitations on modeling process and restricts the application base of such models.

In this dissertation a method is described for creating and rendering convolution surfaces built upon an expanded set of skeletal primitives: points, line segments, polygons, curves and planes. Analytical solutions to the convolution integral are presented for a number of implicit primitives and potential functions. A comparative analysis for a number of convolution kernels is given.

In addition to conventional techniques, commonly used in implicit modeling, this dissertation describes a set of new modeling methods, which offer a better flexibility for modeling with implicit surfaces.

Finally, an efficient ray-tracing algorithm is presented, which is capable of producing high-quality images of objects modeled with convolution surfaces. The algorithm outperforms all ray-tracing algorithms in its class known to date.

Developing, modeling and rendering issues, discussed in this dissertation are illustrated by original images developed and rendered by the author.

Statement

Most parts of this dissertation appeared as technical reports published by the Department of Computer Science [42, 62, 63] and the School of Computer Science and Software Engineering [65] at Monash University, Australia. Additionally, the following papers, based on the material presented in the dissertation, have appeared or are to appear:

- “Kernel functions in convolution surfaces: a comparative analysis”, accepted for publication in *The Visual Computer* in 1999. Based on Chapter 2.
- “Creating and Rendering Convolution Surfaces” (co-authored with Jon McCormack), *Computer Graphics Forum*, 17(2), 1998. Based on Chapter 2.
- “Interactive shape design with convolution surfaces”, accepted for presentation at *Shape Modeling International '99* conference in The University of Aizu, Japan, March 1999. Proceedings of the conference are to be published by IEEE Computer Society Press. Based on Chapter 3.
- “Fast Ray Tracing of Implicit Surfaces”, presented at the Third International Workshop on Implicit Surfaces *Implicit Surfaces '98* in Seattle, USA, and published in the conference proceedings. This paper was voted as the best paper of the conference and was recommended for publication in *Computer Graphics Forum*. Based on Chapter 4.

All software used to produce the pictorial material presented in this dissertation was written by the author. This amounts to nearly 60,000 lines of code.

To the best of my knowledge, this dissertation contains no material which has been accepted for the award of any other degree or diploma in any university or other institution.

Acknowledgments

I am very grateful to Jon McCormack for his advice, guidance and support during the work on this dissertation. His patience and skills in pruning the search space of my research were invaluable.

I am also indebted to Peter Tischer for his advice and discussions and to Jules Bloomenthal for finding time and will to read and correct the early versions of my papers. I am also grateful to John Crossley for his help with proof-reading of my research reports.

My very special thanks to Jim Kajiya for introducing me to the wonderful world of computer graphics in general and ray-tracing in particular.

I want to thank David Albrecht and Alexander Kolesnikov for helping me to sort out things with finite-support functions. Many thanks to Ken Shoemake for his help and discussion on variable arc length in convolution of cubic primitives.

Most of all, I want to thank Katya for her love and encouragement which were the best resources that I enjoyed during these long years of my study.

