

Texturing Surfaces Using Reaction-Diffusion

by
Greg Turk

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
1992

Approved by:

Advisor: Henry Fuchs

Reader: Jonathan Marshall

Reader: Turner Whitted

James Coggins

Albert Harris

© 1992
Greg Turk
ALL RIGHTS RESERVED

GREG TURK. Texturing Surfaces Using Reaction-Diffusion (under the direction of Henry Fuchs).

Abstract

This dissertation introduces a new method of creating computer graphics textures that is based on simulating a biological model of pattern formation known as *reaction-diffusion*. Applied mathematicians and biologists have shown how simple reaction-diffusion systems can create patterns of spots or stripes. Here we demonstrate that the range of patterns created by reaction-diffusion can be greatly expanded by cascading two or more reaction-diffusion systems. Cascaded systems can produce such complex patterns as the clusters of spots found on leopards and the mixture of spots and stripes found on certain squirrels.

This dissertation also presents a method for simulating reaction-diffusion systems directly on the surface of any given polygonal model. This is done by creating a mesh for simulation that is specifically fit to a particular model. Such a mesh is created by first randomly distributing points over the surface of the model and causing these points to repel one another so that they are evenly spaced over the surface. Then a mesh cell is created around each point by finding the Voronoi region for each point within a local planar approximation to the surface. Reaction-diffusion systems can then be simulated on this mesh of cells. The chemical concentrations resulting from such a simulation can be converted to color values to create a texture. Textures created by simulation on a mesh do not have the problems of texture distortion or seams between patches that are artifacts of some other methods of texturing.

Two methods of rendering these synthetic textures are presented. The first method uses a new surface of triangles that closely matches the original model, but whose vertices are taken from the cells of the simulation mesh. These vertices are assigned colors based on the simulation values, and this re-tiled model can be rapidly displayed on a graphics workstation. A higher-quality image can be created by computing each pixel's color value using a weighted average of the chemical concentration at nearby mesh points. Using a smooth cubic weighting function gives a satisfactory reconstruction of the underlying function specified by the values at the mesh points. Several low-pass filtered versions of the texture are used to avoid aliasing when the textured object covers a small portion of the screen. The proper color value at a pixel is found by selecting from the appropriately filtered level.

Acknowledgments

I thank Henry Fuchs for giving me the freedom and the encouragement to pursue the research that led to this dissertation. I am grateful for the guidance that Albert Harris gave to me about issues in developmental biology and for the many pleasant conversations we had on the subject. I thank Turner Whitted for his advice on technical issues in computer graphics, especially during the critical time when I was first preparing an article about this work. I also thank James Coggins and Jonathan Marshall for their excellent suggestions for improving this dissertation.

I thank the many friends that have offered help and encouragement. In particular I thank David Banks, David Ellsworth, Steve Molnar, Carl Mueller, Marc Olano, Mark Parris, Penny Rheingans, and Brice Tebbs. These are the people that made my years at UNC a joy.

I thank Mary McFarlane for the love and patience she has shown through the often difficult process of finishing this dissertation.

Finally, I thank my parents for the love, understanding and encouragement that they have shown throughout my education.

Figure Credits

Some of the figures in this dissertation were made with the generous help of others. I thank all of them.

Special thanks are due to Marc Olano, Carl Mueller, and Andrei State for helping me during the last few weeks before my defense. Marc Olano created Figures 1.6, 1.7, and 1.8. Carl Mueller made Figures 1.2, 1.4, 1.5, 5.7, 5.8, and 5.9. Figure 2.1 was drawn by Andrei State. Figure 5.2 is re-printed from Paul Heckbert's Master's thesis with his permission. Figures 1.3, 1.6, 1.7, and 1.8 were created using a version of Photorealistic RenderMan that was donated to UNC by Pixar (thanks to Tony Apodaca).

Thanks goes to Rhythm & Hues for the horse model that appears in many of the figures. The sea-slug model of Figure 5.13 is courtesy of Apple Computer's Vivarium Program. The giraffe model of Figures 4.14 and 4.15 was created by Steve Speer. The minimal surface in Figures 5.22 and 5.23 was created by James T. Hoffman, and is based on a mathematical description due to Celso Costa, David Hoffman and William Meeks III.

Contents

List of Illustrations	viii
1 Synthetic Texturing	1
1.1 Introduction	1
1.2 Overview of Dissertation	3
1.2.1 Guide to Related Work	4
1.3 A Definition of Texture with Examples	5
1.4 The Three Steps to Texturing (Previous Work)	9
1.4.1 Texture Acquisition	10
1.4.2 Texture Mapping	11
1.4.3 Texture Rendering	12
1.5 An Idealized System	14
2 Pattern Formation in Developmental Biology	16
2.1 Introduction to the Cell	16
2.2 Animal Development	17
2.3 Cell Actions	18
2.3.1 Cell Division	19
2.3.2 Shape Change	20
2.3.3 Migration	21
2.3.4 Materials from Cells: The Cell Matrix, Hormones and Morphogens	21
2.3.5 Cell Death	22
2.3.6 Determination	22
2.4 Information that Guides a Cell	23
2.4.1 Chemical Messages	23
2.4.2 Haptotactic Gradients	24
2.4.3 Electric Fields	24
2.4.4 Structural Information and Mechanical Forces	25
2.4.5 Contact Inhibition	25
2.4.6 Internal State of Cell	25
2.5 Pattern Formation	26
2.5.1 Gradient Model	26
2.5.2 Reaction-Diffusion	28
2.5.3 Patterns Created by Chemotaxis	29
2.5.4 Mechanical Formation of Patterns	29
2.6 Summary	30

3	Reaction-Diffusion Patterns	31
3.1	The Basics of Reaction-Diffusion Systems	31
3.1.1	A Mathematical Description of Reaction Diffusion	32
3.1.2	Simulation of Reaction-Diffusion	34
3.2	Reaction-Diffusion in Biology (Previous Work).....	37
3.3	Cascade Systems	39
3.4	An Interactive Program for Designing Complex Patterns	40
3.4.1	Creating a New Pattern	41
3.4.2	Patterns Created by Cascade	43
4	Simulating Reaction-Diffusion on Polygonal Surfaces	46
4.1	Simulation on Joined Patches (Previous Work)	47
4.2	Requirements for Simulation Meshes	49
4.3	Even Distribution of Points over a Polygonal Surface	50
4.3.1	Random Points Across a Polyhedron	50
4.3.2	Point Relaxation	52
4.4	Generating Voronoi Cells on the Model.....	55
4.4.1	Meshes for Anisotropic Diffusion	58
4.5	Simulation on a Mesh	58
4.6	Pattern Control Across a Surface	61
4.6.1	Stripe Initiation	61
4.6.2	Parameter Specification Using Diffusion	61
4.6.3	Using Curvature to Specify Parameters	64
4.6.4	Future Work in Pattern Control	67
5	Rendering Reaction-Diffusion Textures	68
5.1	Rendering Image-Based Textures (Previous Work)	68
5.1.1	Transformation Between Texture Space and Image Space	69
5.1.2	Aliasing and Filtering	70
5.1.3	Two Texture Sampling Methods: Point Sampling and Mip maps	74
5.2	Interactive Texture Rendering	77
5.3	Using Surface Re-Tiling for Rapid Rendering	78
5.3.1	Re-Tiling using Constrained Triangulation	80
5.3.2	Macro-Displacement Mapping	84
5.4	High-Quality Rendering of Reaction-Diffusion Textures	85
5.4.1	Sparse Interpolation on Polygonal Models	85
5.4.2	Diffusion of Colors for Pre-Filtering	88
5.4.3	Filter Quality	89
5.4.4	Bump Mapping	96
5.5	Hierarchy of Meshes	97
5.5.1	Building Nested Point Sets	97
5.5.2	Sharing Patterns Between Levels in Hierarchies	99
5.5.3	Hierarchies of Polygonal Models	100

6 Conclusion	101
6.1 Summary of Contributions	101
6.2 Future Work	102
Appendix A: Cascade Language	103
Appendix B: Reaction-Diffusion Equations	105
B.1 Turing Spots	106
B.2 Meinhardt Spots	107
B.3 Meinhardt Stripes	109
Appendix C: Cascaded Systems	111
References	117

List of Illustrations

Figure 1.1:	Stripe texture created using reaction-diffusion. Top is an untextured horse and the bottom shows zebra stripes on the same model.	2
Figure 1.2:	Light interacting with perfectly shiny surface.	5
Figure 1.3:	Example of environment mapping. The Utah Teapot reflecting the UNC Old Well.	6
Figure 1.4:	Diagram of how light interacts at a location on a diffuse surface.	7
Figure 1.5:	Using bump mapping to modify surface normals.	7
Figure 1.6:	Example of bump mapping (center) and displacement mapping (right). ...	7
Figure 1.7:	Transparent texturing.	8
Figure 1.8:	Texturing a cubic surface patch.	9
Figure 1.9:	Aliased checkerboard (left) and anti-aliased version (right).	13
Figure 2.1:	Cell division transforms egg into morula and then into blastula.	19
Figure 2.2:	Gastrulation.	20
Figure 2.3:	Formation of the neural tube.	20
Figure 2.4:	<i>Hydra</i>	27
Figure 2.5:	Pattern of chemical concentration from reaction-diffusion system.	27
Figure 3.1:	A row of cells. Molecular bridges allow morphogens to diffuse between neighboring cells.	31
Figure 3.2:	One-dimensional example of reaction-diffusion. Chemical concentration is shown in intervals of 400 time steps.	33
Figure 3.3:	Reaction-diffusion on a square grid. Large spots, small spots, cheetah and leopard patterns.	36
Figure 3.4:	Irregular spots, reticulation, random stripes and mixed large-and-small stripes.	36
Figure 3.5:	User interface of <i>Cascade</i> program.	42
Figure 3.6:	A variety of patterns created by <i>Cascade</i>	44
Figure 4.1:	Isotropic and anisotropic patterns of spots.	48
Figure 4.2:	Choosing random point in triangle. Top: $s + t \leq 1$. Bottom: $s + t > 1$	51
Figure 4.3:	Random points in plane and the same points after relaxation.	52
Figure 4.4:	Mapping nearby point Q onto plane of point P . Left shows when Q is on adjacent polygon. Right shows more remote point Q	54
Figure 4.5:	Amount of diffusion between adjacent cells depends on size of shared boundary.	56
Figure 4.6:	Voronoi regions of the points shown in Figure 4.3.	56
Figure 4.7:	Spot pattern from anisotropic diffusion on test object.	59
Figure 4.8:	Diffusion between cells on square grid (left) and between Voronoi regions (right).	59

Figure 4.9:	Reaction-diffusion on test object. Top shows Voronoi regions, bottom shows chemical concentration as color.	60
Figure 4.10:	Zebra stripes on horse model, created by simulating reaction-diffusion on surface.	62
Figure 4.11:	Stripes are initiated at the head and hooves.	62
Figure 4.12:	Result of flood-fill over horse model from key positions.	63
Figure 4.13:	Values from Figure 4.12 after being smoothed by diffusion.	63
Figure 4.14:	Approximation of curvature over giraffe model. Areas of high curvature are red and areas of low curvature are blue.	65
Figure 4.15:	Size of spots on giraffe model are guided by the curvature estimate shown in Figure 4.14.	65
Figure 4.16:	Curvature along a path in the plane (left) and curvature approximation used at vertices.	66
Figure 5.1:	Mapping from texture to image space.	69
Figure 5.2:	Conceptual model of the processes involved in rendering a texture.	72
Figure 5.3:	Two-dimensional illustration of the rendering process.	73
Figure 5.4:	Pixel's area mapped into texture space (left) and the appropriate filtering computed from the mip map (right).	76
Figure 5.5:	Original polygons of a horse model (top) and a re-tiled version of the same model (bottom).	81
Figure 5.6:	Re-tiled horse model, where vertices are colored based on a reaction-diffusion pattern.	82
Figure 5.7:	Constrained triangulation.	82
Figure 5.8:	Mutual tessellation of a face to incorporate mesh points.	82
Figure 5.9:	Removing an old vertex from a mutual tessellation.	83
Figure 5.10:	Macro-displacement bumpy horse.	84
Figure 5.11:	Weighted average at Q of two nearby sample points.	86
Figure 5.12:	Leopard-horse, rendered using the weighted-average of nearby mesh values.	87
Figure 5.13:	Increasingly blurred versions of a texture on a sea slug.	87
Figure 5.14:	Frames from animation that was anti-aliased using multiple band-limited versions of a stripe texture.	90
Figure 5.15:	Frames from animation with no anti-aliasing.	90
Figure 5.16:	Spatial (top) and frequency (bottom) graph of the cubic weighting function w	92
Figure 5.17:	Spatial (top) and frequency (bottom) graph of a Gaussian filter.	93
Figure 5.18:	Spatial (top) and frequency (bottom) graph of a box filter.	94
Figure 5.19:	Bumpy horse created by perturbing the surface normals based on a reaction-diffusion pattern.	96
Figure 5.20:	A hierarchy of points on a regular grid.	98
Figure 5.21:	An irregular set of nested points in the plane.	98
Figure 5.22:	Nested points on a minimal surface.	99
Figure 5.23:	Three re-tiled versions of a minimal surface with texturing. From left to right the models contain 16000, 4000, and 1000 vertices.	101