



**FLINDERS  
UNIVERSITY**  

---

**ADELAIDE  
AUSTRALIA**

# Trilinear Projection

by

Scott Vallance

B.Sc. (Hons) (Flinders University of South Australia) 1999

A thesis presented to the  
Flinders University of South Australia  
in total fulfillment of the requirements for the degree of  
Doctor of Philosophy

Adelaide, South Australia, 2005  
© (Scott Vallance, 2005)

# Certification

I certify that this thesis does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any university; and that to the best of my knowledge and belief it does not contain any material previously published or written by another person except where due reference is made in the text.

As requested under Clause 14 of Appendix D of the *Flinders University Research Higher Degree Student Information Manual* I hereby agree to waive the conditions referred to in Clause 13(b) and (c), and thus

- Flinders University may lend this thesis to other institutions or individuals for the purpose of scholarly research;
- Flinders University may reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

Signed

Dated

Scott Vallance

# Abstract

In computer graphics a projection describes the mapping of scene geometry to the screen. While linear projections such as perspective and orthographic projection are common, increasing applications are being found for nonlinear projections, which do not necessarily map straight lines in the scene to straight lines on the screen. Nonlinear projections occur in reflections and refractions on curved surfaces, in art, and in visualisation.

This thesis presents a new nonlinear projection technique called a trilinear projection that is based on the trilinear interpolation of surface normals used in Phong shading. Trilinear projections can be combined to represent more complicated nonlinear projections.

Nonlinear projections have previously been implemented with ray tracing, where rays are generated by the nonlinear projections and traced into the scene. However for performance reasons, most current graphics software uses scanline rendering, where a scene point is imaged on a screen as a function of the projection parameters. The techniques developed in this thesis are of this nature.

This thesis presents several algorithms used in trilinear projection:

1. An algorithm to analytically determine which screen locations image a given scene point.
2. An algorithm that correctly connects projected vertices. Each scene point may be imaged multiple times, which means a projected scene triangle may form from one to four different shapes of from two to nine vertices. Once connected, the projected shapes may be rendered with standard scanline algorithms.
3. An algorithm to more accurately render the curved edges between projected vertices.
4. A scene-space edge-clipping algorithm that handles continuity issues for projected shapes across composite projections.

The trilinear projection technique is demonstrated in two different application areas: visualisation, and reflections and refractions. Specifically, various nonlinear projections that are congruent with pre-existing visualisation techniques are implemented with trilinear projections and a method for approximating the reflections and refractions on curved surfaces with trilinear projections is presented. Finally, the performance characteristics of the trilinear projection is explored over various parameter ranges and compared with a naive ray tracing approach.

# Contents

<b>List of Figures</b>	<b>vii</b>
<b>List of Tables</b>	<b>x</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Nonlinear Projection . . . . .	1
1.1.1 Artistic Nonlinear Projection . . . . .	2
1.1.2 Strip Cameras . . . . .	2
1.1.3 Multi-Perspective Images as a Basis for Resynthesis . . . . .	4
1.1.4 Visualisation with Nonlinear Projection . . . . .	4
1.2 Ray tracing and Scanline Rendering . . . . .	6
1.3 Thesis Scope . . . . .	6
1.4 Thesis Overview . . . . .	7
<b>2 Nonlinear Projection Surfaces</b>	<b>9</b>
2.1 Projection Surfaces . . . . .	9
2.2 Mesh Surfaces and Surface Continuity . . . . .	10
2.3 Representing a Curved Surface with a Trilinear Interpolated Triangle . . . . .	11
2.3.1 Interpolation . . . . .	11
2.3.2 Phong Shading . . . . .	12
2.4 Ray Tracing with a Trilinear Projection . . . . .	13
2.5 Summary . . . . .	13
<b>3 Projecting a Point with a Trilinear Projection</b>	<b>15</b>
3.1 Projecting a Point with a Trilinear Projection . . . . .	16
3.2 Treating the Trilinear Projection as a Parametric Triangle . . . . .	16
3.3 Determining the Coplanarity of the Parametric Triangle and the Scene Point	17
3.4 Barycentric Coordinate Conversion . . . . .	18
3.5 Multiple Solutions . . . . .	20
3.6 Precalculating Partial Coefficient Values . . . . .	20
3.7 Parametric Triangle Sides and Containment . . . . .	21
3.8 Algorithm for Projecting a Scene Point . . . . .	24
3.9 Summary . . . . .	25

<b>4</b>	<b>Projecting a Triangle with a Trilinear Projection</b>	<b>26</b>
4.1	Drawing a Scene Triangle . . . . .	26
4.2	Determining Shapes . . . . .	27
4.3	Drawing Shapes . . . . .	28
4.3.1	An Algorithm to Connect Vertex Lists . . . . .	31
4.4	Example Shape Images . . . . .	32
4.5	Tessellation . . . . .	37
4.5.1	Scene Triangle Tessellation . . . . .	37
4.5.2	Parametric Triangle Slices . . . . .	38
4.6	Viewing Plane Intersection . . . . .	40
4.7	Summary . . . . .	42
<b>5</b>	<b>Multiple Surface Triangles</b>	<b>44</b>
5.1	Screen Space Clipping . . . . .	44
5.2	Scene Space Clipping . . . . .	45
5.2.1	Algorithm for Determining Intersection $t$ Values . . . . .	46
5.2.2	Integrating Intersection Points into Drawing Primitives . . . . .	47
5.3	Summary . . . . .	50
<b>6</b>	<b>Nonlinear Projection for Visualisation</b>	<b>51</b>
6.1	Detail and Context . . . . .	51
6.2	Multiple Perspective Views . . . . .	53
6.3	Mappings . . . . .	56
6.4	Summary . . . . .	58
<b>7</b>	<b>Reflections and Refractions</b>	<b>60</b>
7.1	Integrating Reflection and Refraction Projections into a Scene . . . . .	60
7.2	Reflections . . . . .	60
7.2.1	First-Hit Reflections on a Polygon Mesh . . . . .	61
7.2.2	Approximated First-Hit Reflections . . . . .	62
7.2.3	Multi-Hit Reflections on a Polygon Mesh . . . . .	63
7.3	Refraction . . . . .	63
7.3.1	First-hit Refraction on a Polygon Mesh . . . . .	64
7.3.2	Approximating Refraction . . . . .	64
7.3.3	Multi-Hit Refraction on a Polygon Mesh . . . . .	64
7.4	Example Projections . . . . .	65
7.5	Summary . . . . .	66
<b>8</b>	<b>Performance Evaluation</b>	<b>70</b>
8.1	Experimental Conditions . . . . .	70
8.1.1	Data . . . . .	71
8.1.2	Caveats . . . . .	71
8.2	Ray Tracing . . . . .	72
8.3	Trilinear Projection . . . . .	74
8.4	Speed up . . . . .	76

8.5	Tessellation Methods . . . . .	76
8.6	Multiple Trilinear Projections . . . . .	80
8.7	Nonlinear Projection for Visualisation, and Reflections and Refractions . . . . .	82
8.8	Summary . . . . .	84
<b>9</b>	<b>Related Work</b>	<b>85</b>
9.1	Ray Tracing . . . . .	85
9.1.1	Beam Tracing . . . . .	86
9.1.2	Spatial Subdivision . . . . .	87
9.1.3	Hardware Ray Tracing . . . . .	87
9.2	Ray Tracing Nonlinear Projections . . . . .	88
9.2.1	Ray Tracing with Extended Cameras . . . . .	88
9.2.2	Cubism and Cameras: Free-form Optics for Computer Graphics . . . . .	89
9.2.3	Multi-Perspective Images for Visualisation . . . . .	90
9.2.4	General Linear Cameras . . . . .	90
9.3	Scanline Rendering . . . . .	91
9.3.1	Multi-Pass Rendering . . . . .	91
9.3.2	Reflections on Spheres and Cylinders of Revolution . . . . .	92
9.3.3	Multiple-Center-of-Projection Images . . . . .	93
9.4	Object Distortion for Nonlinear Projections . . . . .	93
9.4.1	Distortion Methods for Visualisation . . . . .	93
9.4.2	Interactive Reflections on Curved Objects . . . . .	95
9.4.3	Specular Path Perturbation . . . . .	95
9.4.4	Region of Influence Cameras . . . . .	96
9.5	Approximating Reflections on Curved Objects with Image Based Rendering . . . . .	96
9.5.1	Environment Mapping . . . . .	97
9.5.2	Extended Environment Mapping . . . . .	97
9.5.3	Parameterized Environment Maps . . . . .	98
9.5.4	Light Field Rendering . . . . .	99
9.6	Summary . . . . .	99
<b>10</b>	<b>Conclusion</b>	<b>101</b>
10.1	Summary . . . . .	101
10.2	Contributions . . . . .	101
10.2.1	Projecting a Scene Point with Trilinear Projection . . . . .	102
10.2.2	Projecting a Scene Triangle with Trilinear Projection . . . . .	102
10.2.3	Parametric Triangle Slicing . . . . .	103
10.2.4	Scene Space Clipping . . . . .	103
10.2.5	The Application of Trilinear Projection in Visualisation . . . . .	103
10.2.6	The Application of Trilinear Projection in Rendering Reflections and Refractions on Curved Surfaces . . . . .	104
10.3	Further Work . . . . .	104
10.4	Conclusion . . . . .	105

<b>A Expanded Equations</b>	<b>106</b>
A.1 Parametric Triangle and Scene Point Coplanarity Test . . . . .	106
A.2 Line Segment Intersection Coplanarity Cubic . . . . .	107
A.3 Precalculation for Cubic Coefficients . . . . .	109
<b>B Vector Properties</b>	<b>110</b>
<b>C View and Scene Data</b>	<b>111</b>
<b>D Tabulated Performance Results</b>	<b>114</b>
D.1 Ray Tracing Results . . . . .	114
D.2 Trilinear Projection Results . . . . .	118
D.3 Trilinear Projection with Scene Space Clipping Results . . . . .	121
D.4 Ray Tracing on Different Configurations . . . . .	124
D.5 Trilinear Projection on Different Configurations . . . . .	126
D.6 Trilinear Projection with Scene Triangle Tessellation Results on Different Configurations . . . . .	128
D.7 Trilinear Projection with Parametric Triangle Slicing Results on Different Configurations . . . . .	131
<b>E Context in Planar 3D Navigation</b>	<b>134</b>
<b>F Multi-Perspective Images for Visualisation</b>	<b>142</b>
<b>G Inward Looking Projections</b>	<b>151</b>
<b>Bibliography</b>	<b>156</b>



# List of Figures

1.1	“Fishermans Evening Song” by Xu Daoning, circa 11th Century . . . . .	2
1.2	“High and Low” by M. C. Escher, an example of a nonlinear projection . . .	3
1.3	A strip camera photograph of a man’s head [Dav01] . . . . .	4
1.4	A multi-perspective image for use in image resynthesis [Chu01] . . . . .	5
1.5	A multi-perspective image for use in image resynthesis [WFH <sup>+</sup> 97] . . . . .	5
1.6	A child’s depiction of a cube, subsequently redrawn [Wil97] . . . . .	6
2.1	Projection surfaces (top view): (a) discontinuity of ray directions on a pro- jection surface defined by perspective projections and (b) continuity of ray directions on a shared-normal interpolated projection surface . . . . .	10
2.2	Triangle with arbitrary normal vectors . . . . .	11
2.3	Triangle with interpolated normal vectors . . . . .	12
2.4	A Phong shaded cube: (a) normal vectors perpendicular to the faces (b) normals coincident with the cube centre . . . . .	13
2.5	Ray tracing a trilinear projection with rasterising . . . . .	14
3.1	Interpolated surface ray intersecting a scene point . . . . .	15
3.2	A parametric triangle shown at different values of $t$ . . . . .	17
4.1	A (2,2,2,3) shape configuration: (a) ray trace (b) trilinear projection . . . . .	34
4.2	A (3,3,3) shape configuration: (a) ray trace (b) trilinear projection . . . . .	34
4.3	A (4,2,3) shape configuration: (a) ray trace (b) trilinear projection . . . . .	34
4.4	A (4,5) shape configuration: (a) ray trace (b) trilinear projection . . . . .	35
4.5	A (6,3) shape configuration: (a) ray trace (b) trilinear projection . . . . .	35
4.6	A (2,7) shape configuration: (a) ray trace (b) trilinear projection . . . . .	35
4.7	A (9) shape configuration: (a) ray trace (b) trilinear projection . . . . .	36
4.8	Example of the error inherent in the linear approximation of curved shapes	37
4.9	Scene triangle tessellated into 25 triangles approximating a (4,5) shape con- figuration . . . . .	38
4.10	Scene triangle sampled at 5 extra $t$ levels per shape approximating a (4,5) shape configuration . . . . .	39
4.11	A shape partially behind the viewing plane . . . . .	40
5.1	A scene triangle spanning two trilinear projections with a discontinuity . . .	45

5.2	A trilinear projection edge swept out into scene space and intersected with a scene triangle . . . . .	46
6.1	A Distortion-Oriented Display mesh projection surface . . . . .	52
6.2	A Distortion-Oriented Display projection surface and a cube scene . . . . .	53
6.3	A perspective projection of a cube . . . . .	54
6.4	A Distortion-Orientation projection of a cube (a) ray trace (b) trilinear projection . . . . .	54
6.5	A maze distorted in a cylindrical fashion to show context . . . . .	55
6.6	A multiple-perspective approach to showing first person detail and side view context . . . . .	56
6.7	A cube rendered from a surface derived from Figure 6.6 (a) ray trace (b) trilinear projection . . . . .	57
6.8	An spherical mesh mapping a relation between the scene data and surface . . . . .	58
6.9	A cube rendered by a spherical projection surface (a) ray trace (b) trilinear projection . . . . .	59
7.1	A diagram of a first-hit reflection . . . . .	61
7.2	Error in an approximation of first-hit reflections . . . . .	63
7.3	A cube reflected in a sphere . . . . .	65
7.4	A cube reflected on a sphere: (a) ray traced, (b) 1x1 surface, (c) 2x2 surface, (d) 3x3 surface, (e) 4x4 surface, (f) 5x5 surface . . . . .	67
7.5	A cube reflected on a plane with perturbed normals, implemented as a 5x5 trilinear projection surface: (a) ray traced (b) trilinear projection . . . . .	68
7.6	A cube refracted through a plane with spherical normals: (a) ray traced, (b) 1x1 surface, (c) 2x2 surface, (d) 3x3 surface, (e) 4x4 surface, (f) 5x5 surface . . . . .	69
8.1	Execution time versus resolution for ray tracing different configurations . . . . .	72
8.2	Execution time versus resolution for ray tracing across different complexity scenes . . . . .	73
8.3	Execution time versus number of scene triangles for ray tracing across different resolutions . . . . .	73
8.4	Execution time versus resolution for trilinear projecting different configurations . . . . .	74
8.5	Execution time versus resolution for trilinear projection across different complexity scenes . . . . .	75
8.6	Execution time versus number of scene triangles for trilinear projection across different resolutions . . . . .	75
8.7	Relative speedup versus resolution for trilinear projection across different configurations . . . . .	76
8.8	Relative speedup versus resolution for trilinear projection across different complexity scenes . . . . .	77
8.9	Relative intensity of the difference mask versus tessellation factor for parametric triangle slicing over different configurations . . . . .	78
8.10	Relative intensity of the difference mask versus tessellation factor for scene triangle tessellation over different configurations . . . . .	78

8.11	Relative intensity of the difference mask versus tessellation factor averaged over each configurations for parametric triangle slicing and scene triangle tessellation . . . . .	79
8.12	Execution time versus tessellation factor averaged over each configurations for parametric triangle slicing and scene triangle tessellation . . . . .	79
8.13	Execution time versus number of trilinear projections for ray tracing across different resolutions . . . . .	80
8.14	Execution time versus number of trilinear projections across different resolutions . . . . .	81
8.15	Execution time versus number of trilinear projections across different resolutions . . . . .	81
8.16	Relative execution time for clipped and non-clipped trilinear projection versus number of trilinear projections across different complexity scenes . . . .	82
9.1	A conventionally rendered set of columns . . . . .	88
9.2	Columns rendered from a torus surface . . . . .	89
9.3	A hand-drawn nonlinear projection of a street scene [Gla00] . . . . .	90
9.4	A cube rendered from a hemisphere surface [VC01b] . . . . .	91
9.5	Catacaustic of the reflection congruence [Gla99] . . . . .	92
9.6	A nonlinear projection of an elephant [RB98] . . . . .	94

# List of Tables

4.1	Possible shape configurations . . . . .	29
8.1	Execution time for rendering examples in this thesis . . . . .	83
B.1	Vector properties of the trilinear projection . . . . .	110
C.1	. . . . .	112
C.2	. . . . .	113
D.1	Ray tracing results over random scene data . . . . .	117
D.2	Trilinear projection results over random scene data . . . . .	120
D.3	Trilinear projection with clipping results over random scene data . . . . .	123
D.4	Ray tracing results on a 2,2,2,3 configuration example . . . . .	124
D.5	Ray tracing results on a 2,3 configuration example . . . . .	124
D.6	Ray tracing results on a 3,3,3 configuration example . . . . .	124
D.7	Ray tracing results on a 4,2,3 configuration example . . . . .	124
D.8	Ray tracing results on a 4,5 configuration example . . . . .	124
D.9	Ray tracing results on a 6,3 configuration example . . . . .	124
D.10	Ray tracing results on a 9 configuration example . . . . .	125
D.11	Trilinear projection results on a 2,2,2,3 configuration example . . . . .	126
D.12	Trilinear projection results on a 2,3 configuration example . . . . .	126
D.13	Trilinear projection results on a 3,3,3 configuration example . . . . .	126
D.14	Trilinear projection results on a 4,2,3 configuration example . . . . .	126
D.15	Trilinear projection results on a 4,5 configuration example . . . . .	126
D.16	Trilinear projection results on a 6,3 configuration example . . . . .	126
D.17	Trilinear projection results on a 9 configuration example . . . . .	127
D.18	Trilinear projection scene triangle tessellation results on a 2,2,2,3 configuration example . . . . .	128
D.19	Trilinear projection scene triangle tessellation results on a 2,3 configuration example . . . . .	128
D.20	Trilinear projection scene triangle tessellation results on a 3,3,3 configuration example . . . . .	129
D.21	Trilinear projection scene triangle tessellation results on a 4,2,3 configuration example . . . . .	129

D.22 Trilinear projection scene triangle tessellation results on a 4,5 configuration example . . . . .	130
D.23 Trilinear projection scene triangle tessellation results on a 6,3 configuration example . . . . .	130
D.24 Trilinear projection scene triangle tessellation results on a 9 configuration example . . . . .	130
D.25 Trilinear projection with parametric triangle slicing results on a 2,2,2,3 configuration example . . . . .	131
D.26 Trilinear projection with parametric triangle slicing results on a 2,3 configuration example . . . . .	131
D.27 Trilinear projection with parametric triangle slicing results on a 3,3,3 configuration example . . . . .	132
D.28 Trilinear projection with parametric triangle slicing results on a 4,2,3 configuration example . . . . .	132
D.29 Trilinear projection with parametric triangle slicing results on a 4,5 configuration example . . . . .	133
D.30 Trilinear projection with parametric triangle slicing results on a 6,3 configuration example . . . . .	133
D.31 Trilinear projection with parametric triangle slicing results on a 9 configuration example . . . . .	133

# List of Listings

3.1	Projecting a scene point onto a parametric triangle . . . . .	25
4.2	Sorting vertex lists into shapes . . . . .	28
4.3	Sorting vertex lists into shapes and ordering by connectivity . . . . .	32
4.4	Sampling with discrete parametric triangle slices . . . . .	40
4.5	Calculating view-plane intersections . . . . .	42
5.6	Finding edge intersection $t$ values . . . . .	47
5.7	Finding clipping points in shapes . . . . .	48
5.8	Inserting clipping points into shapes . . . . .	49

## Acknowledgements

This work was supported by a Flinders University Research Scholarship for which I am very grateful. None of this would have been possible without the support of a great many people whom I would like to thank. My partner, Fran, for being there when I needed it most, thank you. My family, who supported me in a myriad of ways, from their love down to the practical everyday things. My supervisor, Paul, for his endless advice and enjoyable research discussions. All my fellow research slaves, aka PhD candidates, for their camaraderie, and especially Aaron and Ron for always being there for a chat. To my friends whom I have neglected too much whilst writing this thesis, I apologise.