



Copyright Statement

The digital copy of this thesis is protected by the Copyright Act 1994 (New Zealand). This thesis may be consulted by you, provided you comply with the provisions of the Act and the following conditions of use:

- Any use you make of these documents or images must be for research or private study purposes only, and you may not make them available to any other person.
- Authors control the copyright of their thesis. You will recognise the author's right to be identified as the author of this thesis, and due acknowledgement will be made to the author where appropriate.
- You will obtain the author's permission before publishing any material from their thesis.

To request permissions please use the Feedback form on our webpage.
<http://researchspace.auckland.ac.nz/feedback>

General copyright and disclaimer

In addition to the above conditions, authors give their consent for the digital copy of their work to be used subject to the conditions specified on the Library [Thesis Consent Form](#)

Practical Light Field Rendering

Jarno van der Linden

A thesis submitted in fulfilment of the
requirements for the degree of
Doctor of Philosophy

University of Auckland
July 26, 2005



Abstract

Light field rendering is an image-based rendering method first explicitly described in 1996. The light field function determines the radiance, or colour, seen along any ray into or out of a scene. It is a function over a 4D space of directed lines. With some restrictions on the placement of the eyepoint, almost any view of the scene can be recreated by sampling the light field function for every ray that goes through the eyepoint and the scene. In practice, the light field function consists of discrete samples obtained either photographically using a calibrated camera setup, or from computer-generated imagery. Light field rendering is a purely image-based rendering technique. No geometric information about the scene is required or derived in the process.

Although potentially capable of creating photorealistic renderings quickly, light field rendering has seen little use in either the research community, or the wider graphics world. The reasons may be traced to a number of weaknesses and limitations in the method, which make its use impractical.

This thesis explores and improves upon several aspects of light field rendering, with the aim of making it a more practical means of image-based rendering. The main contributions are:

- Better interpolation using a focal distance for improved rendering quality.
- Incorporating depth information in the light field.
- A light field parameterisation which encompasses the entire light field scene within a single structure.
- Creation and rendering of compressed light fields using the standard MPEG video encoding scheme.
- Use of light fields as a scene element in combination with polygon rendering.

The most important contribution of this thesis is the use of light field rendering in combination with another rendering method, namely polygon rendering. A scene can be composed using both polygon objects and light fields. Light fields can intersect with other light fields or with polygonal objects, and still be rendered in a consistent manner from all view directions. This demonstrates, for the first time, that light fields can be used as part of a general rendering system in a practical manner.

Thanks

First of all I thank my supervisor Dr. Richard Lobb. I'm sure this thesis has taken quite a bit longer than any of us anticipated, so I'm grateful you let me do my thing without getting on my back too much.

Many thanks also go out to all the members of the Graphics Group past and present. Not only have they given me different insights into my own work, but have also exposed me to many different corners in the weird and wonderful world of computer graphics.

I thank the Internet for being an endless source of information. I also curse the Internet for being an *endless* source of information.

My parents, of course, are due an infinite amount of gratitudes for their unbounded love and support.

Lastly I like to thank myself, for sticking with it even when the task looked never-ending.

Contents

Abstract	v
Thanks	vii
Contents	ix
List of Figures	xv
List of Tables	xix
1 Introduction	1
1.1 Primitives	1
1.2 Polygon Modelling and Rendering	1
1.3 Enhancement through images	2
1.4 Image-based modelling and rendering	2
1.5 Plenoptic Modelling	5
1.6 Light Fields	6
1.7 Other work on Light Fields	8
1.7.1 Light slab mathematics	8
1.7.2 Changing illumination of light fields	8
1.7.3 Alternative parameterisations	9
1.7.4 Illumination with light fields	10
1.7.5 Editing of light fields	11
1.7.6 Surface light fields	12
1.8 About the thesis	12
2 Light Field Rendering	15
2.1 Introduction	15
2.2 Acquired and generated light fields	15
2.3 The light slab parameterisation	16
2.4 Light field data structure	17

2.5	Creating a light slab	19
2.6	Light field rendering approaches	23
2.6.1	Light field rendering through raytracing	23
2.6.2	Texturemapped light field rendering	24
2.6.3	Light field rendering by post-processing	25
2.7	Resampling the light field	28
2.7.1	Nearest Neighbour resampling	28
2.7.2	Bilinear (u, v) Interpolation	29
2.7.3	Bilinear (s, t) interpolation	30
2.7.4	UV versus ST resolution	30
2.8	Conclusion	31
3	Focusing in Light Fields	35
3.1	Motivation	35
3.2	Focus in light fields	35
3.3	UVd depth determination	38
3.4	Results	38
3.5	Light fields and optics	41
3.5.1	Thin convex lenses	41
3.5.2	Depth of field	45
3.6	Conclusion	46
4	Image Flow in Light Fields	47
4.1	Motivation	47
4.2	Problems and approach	48
4.3	Related work	48
4.4	Image flow in light fields	49
4.5	Light field rendering using image flow	51
4.5.1	Image flow for interpolation	51
4.5.2	Scan backward	51
4.5.3	Flow forward	52
4.5.4	Tearing	55
4.6	Implementation	56
4.6.1	Reconstruction	56
4.6.2	Scan backward implementation issues	58
4.6.3	Flow forward implementation issues	58
4.7	Results	61
4.7.1	Scan backward	61
4.7.2	Flow forward	62
4.7.3	Specular highlights	62
4.8	Conclusion	63

5	Canonical View Encoding	67
5.1	Motivation	67
5.2	Problems and approach	68
5.3	Related work	68
5.4	The Canonical View	69
5.4.1	Constructing the canonical view	69
5.4.2	Generating a new view	70
5.5	Rendering issues	71
5.5.1	Reconstruction	71
5.5.2	Occlusion	72
5.5.3	Progressive rendering	73
5.6	Canonical view compression	73
5.6.1	Redundancy	73
5.6.2	Tolerance regions	74
5.6.3	The compression problem	74
5.6.4	Compression algorithms	76
5.7	Results	76
5.7.1	Comparison between greedy and FCFS	76
5.7.2	Effectiveness of compression	78
5.8	Conclusion	82
6	The Spiral Sphere-Plane Parameterisation	83
6.1	Motivation	83
6.2	Problems and approach	84
6.3	The sphere-plane parameterisation	85
6.4	Creating sphere-plane light fields	86
6.5	Sampling strategy	86
6.5.1	The Generalised Spiral Set	87
6.5.2	Distribution of GSS points	87
6.6	Rendering with the sphere-plane parameterisation	89
6.6.1	Constructing a new view	89
6.6.2	Mathematics of sphere-plane light field rendering	90
6.7	Implementation	92
6.7.1	Using nearest samples	92
6.7.2	Determining nearest samples	93
6.7.3	Interpolation	94
6.7.4	Reducing computational cost	94
6.7.5	Orthographic rendering	98
6.8	Results	98
6.9	Conclusion	98

7	MPEG-Encoded Light Fields	101
7.1	Motivation	101
7.2	Problems and approach	101
7.3	Related work	102
7.4	MPEG video compression: a primer	103
7.4.1	A hierarchy of layers	103
7.4.2	Macroblocks	103
7.4.3	DCT coding and quantisation	104
7.4.4	DC component differencing	105
7.4.5	RLA and Variable-length encoding	105
7.4.6	Motion estimation	105
7.4.7	Types of frames	106
7.5	Encoding light fields as MPEG video	106
7.5.1	Solving the access problem	107
7.5.2	Placing restrictions	107
7.5.3	Macroblock index and quantiser scale tables	108
7.5.4	DC component differential	108
7.6	Using MPEG compressed light fields	108
7.7	Transparency	109
7.7.1	Luminance versus alpha	111
7.8	Implementation	112
7.9	Results	113
7.10	Conclusion	116
8	Combining Light Field and Polygon Rendering	119
8.1	Motivation	119
8.2	Problems and approach	120
8.3	Light fields in polygon scenes	120
8.4	Background removal	121
8.5	Impostors	122
8.6	Intersecting Light Fields	122
8.6.1	Micro-surfaces	123
8.6.2	Intersecting light fields with micro-surfaces	127
8.6.3	Some simple examples	132
8.6.4	An approximation to light field intersection	137
8.7	Implementation	137
8.7.1	Light field intersection implementation	138
8.7.2	Rendering the light field slices	141
8.7.3	Shadows	143
8.7.4	Light fields intersecting with terrain	144
8.7.5	Multi-pass rendering algorithm	145
8.8	Error analysis	146
8.8.1	Statistical error analysis	150

8.9 Results	152
8.10 Conclusion	154
9 Conclusions	157
A When is orthographic good enough?	163
A.1 Introduction	163
A.2 Limited texture resolution	164
A.3 Limited display resolution	166
A.4 Limited light field resolution	167
A.5 Summary	169
Bibliography	171

List of Figures

1.1	Image flow due to camera movement	4
1.2	The ordering of points A and B as seen from the camera is determined by their relative position as projected on the source cylinder	6
2.1	Light slab representation of a light field	17
2.2	Light slab camera and scene spaces	17
2.3	The field of view of a light slab	18
2.4	Acquiring a light field	20
2.5	Algorithmic overview of creating a light slab	21
2.6	Generating a light field using a sheared perspective	22
2.7	Algorithmic overview of creating a light slab with sheared perspective	23
2.8	Rendering a view of a light slab	24
2.9	Light field rendering using a post-processing method	26
2.10	Algorithm for light field rendering through post-processing	27
2.11	Nearest Neighbour	29
2.12	Bilinear UV	29
2.13	The mosaic effect of nearest neighbour resampling	29
2.14	Bilinear UV , 128×128 ST resolution	30
2.15	Bilinear UV and ST , 128×128 ST resolution	30
2.16	Quantisation during resampling can cause apparent scan direction to change	31
2.17	The effects of using different UV and ST resolutions	32
3.1	Simple bilinear interpolation causes the rays associated with the known neighbour light field samples to diverge	36
3.2	The effect of different focus distances d	37
3.3	Renderings of light fields focused with minimum disparity	39
3.4	Dynamic change of focus as view moves from near to far object	40
3.5	An ideal thin convex lens focuses parallel rays to a point at a given focal length f from the lens	42

3.6	A thin convex lens as a light field	43
3.7	Relationship between coordinates of two light fields separated by a distance d	44
4.1	The relation between flow and depth of a point P in the scene	50
4.2	The relation between UV and ST light field intersection points for some camera position	53
4.3	With tearing, shown in green	57
4.4	No tearing visible, using reconstruction	57
4.5	Scan backward pseudo-code	59
4.6	Flow forward pseudo-code	60
4.7	OpenGL rendering of the tricycle scene	61
4.8	Light field rendered using quadrilinear colour interpolation	61
4.9	Result of the scan backward method	62
4.10	Result of the scan backward method with hole filling	62
4.11	Result of the flow forward method	63
4.12	Flow forward, using $\frac{1}{9}$ th of the samples	63
4.13	Rendering from a teapot light field with specular highlights	64
5.1	Example of tearing for a view with a camera off the UV plane	72
5.2	Scene rendered using the reconstruction method, with filter size corrected for tearing	72
5.3	Some samples and their associated tolerance regions plotted in parameter space	75
5.4	Comparison between Greedy and FCFS compression for the tricycle scene sampled at $4 \times 4 \times 256 \times 256$	77
5.5	Tricycle model, rendered using OpenGL	78
5.6	A view of the tricycle computed from a canonical view encoded light field	78
5.7	Lattice model	79
5.8	A view of the lattice computed from a canonical view encoded light field	79
5.9	Number of compressed samples versus the number of sample points in UV	80
5.10	Compression ratio is linearly related to the number of UV sample points	81
6.1	Geometry of the sphere-plane parameterisation	85
6.2	GSS versus subdivision comparison	88
6.3	Camera parameters used for sphere-plane light fields	90
6.4	Computing the arc-tangent using a lookup table containing one octant	97
6.5	Renderings of a sphere-plane parameterised light field of a tree	99

7.1	Overview of creating and using MPEG encoded light fields	102
7.2	Basic structure of an MPEG video stream	104
7.3	Algorithm for decoding MPEG encoded light fields	110
7.4	A novel view of the Buddha dataset compressed at various bit-rates	114
7.5	Some views generated from the MPEG compressed fish dataset . .	117
7.6	Tree with partial transparencies	118
7.7	Closeup rendering of tree	118
8.1	Problems with intersecting light fields	124
8.2	Micro-surfaces can be thought of as a partial cross-section split into pieces	125
8.3	Schematic of a beam passing through two light fields	127
8.4	A light field is rendered as a number of rectangular slices	138
8.5	Each stack of slices is oriented parallel to the image plane	138
8.6	Regions of space represented by slices	139
8.7	Two light field instances intersecting using slices	140
8.8	Adding contributions of slices may result in missed segments . . .	141
8.9	Applying a shadow visually anchors an object such as a tree to a place in a scene such as a terrain	143
8.10	Multi-pass rendering stages to combine polygon and light field ren- dering	147
8.11	Contour plot of maximum error in final alpha	148
8.12	Contour plots of error in contribution	149
8.13	Contour plots of error in colour contribution	151
8.14	Error distribution for random light fields configurations	153
8.15	Island flyover images	155
A.1	Difference between projection of a point within a light field using perspective and orthographic methods, texture resolution limited .	164
A.2	Difference between projection of a point within a light field using perspective and orthographic methods, display resolution limited .	166
A.3	Difference between projection of a point within a light field using perspective and orthographic methods, light field resolution limited	168
A.4	d_{\max} for a typical light field rendering under various limiting con- ditions	169

List of Tables

5.1	Comparison between the greedy and FCFS methods for various values of global tolerance v	76
5.2	Comparison between greedy and FCFS with similar compression ratios	77
7.1	Sizes and error measurements for the Buddha light field at various target bit-rates	114
7.2	Comparison between MPEG and vector quantisation compression	115
7.3	Averaged rendering times and cache hits for viewing of MPEG compressed light fields	115
8.1	Contrasts between light field as scene and as scene component . .	120
8.2	Cache setup used	143
8.3	Range of random light field parameters	150

