

THE UNIVERSITY OF ADELAIDE

---

School of Computer Science

# Reconstructing 3D Geometry from Multiple Images via Inverse Rendering

John William Bastian

December, 2007

SUBMITTED FOR THE DEGREE OF DOCTOR OF PHILOSOPHY IN THE  
FACULTY OF ENGINEERING, COMPUTER & MATHEMATICAL SCIENCES

## ABSTRACT

An image is a two-dimensional representation of the three-dimensional world. Recovering the information which is lost in the process of image formation is one of the fundamental problems in Computer Vision. One approach to this problem involves generating and evaluating a succession of surface hypotheses, with the best hypothesis selected as the final estimate. The fitness of each hypothesis can be evaluated by comparing the reference images against synthetic images of the hypothesised surface rendered with the reference cameras.

An infinite number of surfaces can recreate any set of reference images, so many approaches to the reconstruction problem recover the largest from this set of surfaces. In contrast, the approach we present here accommodates prior structural information about the scene, thereby reducing ambiguity and finding a reconstruction which reflects the requirements of the user. The user describes structural information by defining a set of primitives and relating them by parameterised transformations. The reconstruction problem then becomes one of estimating the parameter values that transform the primitives such that the hypothesised surface best recreates the reference images.

Two appearance-based likelihoods which measure the hypothesised surface against the reference images are described. The first likelihood compares each reference image against an image synthesised from the same viewpoint by rendering a projection of a second image onto the surface. The second likelihood finds the 'optimal' surface texture given the hypothesised scene configuration. Not only does this process maximise photo-consistency with respect to all reference images, but it prohibits incorrect reconstructions by allowing the use of prior information about occlusion. The second likelihood is able to reconstruct scenes in cases where the first is biased.

## DECLARATION

This work does not contain material that has been accepted for the award of any other degree or diploma in any University or other tertiary institution. 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 has been made in the text.

I give consent that this Dissertation to be made available by the University Library for loan and photo-copying.

I acknowledge that copyright of published works contained within this thesis (as listed below) resides with the copyright holder/s of those works.

BASTIAN, J. W. AND VAN DEN HENGEL, A. J. , Computing image-based re-projection error on graphics hardware, *Proceedings of the Seventh International Conference on Digital Image Computing: Techniques and Applications, (DICTA 2005)*, Cairns, Australia, 6-8 December 2005, IEEE Computer Society 2005

BASTIAN, J. W. AND VAN DEN HENGEL, A. J. , Computing image-based re-projection error on graphics hardware, *Proceedings of the Sixth International Conference on Digital Image Computing: Techniques and Applications, (DICTA 2003)*, Macquarie University, Sydney, Australia, 10-12 December 2003, CSIRO Publishing 2003

# TABLE OF CONTENTS

<b>List of Figures</b>	<b>v</b>
<b>List of Algorithms</b>	<b>xi</b>
<b>Notation</b>	<b>xv</b>
<b>Chapter I: Reconstructing Surfaces from Images</b>	<b>1</b>
1.1 From features to generative models . . . . .	2
1.2 Fitting geometry to images via inverse rendering . . . . .	4
1.3 Outline . . . . .	5
<b>Chapter II: Prior Work on Surface Reconstruction</b>	<b>7</b>
2.1 Geometry-based reconstruction . . . . .	8
2.2 The visual hull . . . . .	9
2.2.1 Computing visual hulls on graphics hardware . . . . .	11
2.3 The photo-hull . . . . .	12
2.3.1 Feature-based refinements . . . . .	13
2.3.2 Space carving . . . . .	18
2.3.3 Visibility . . . . .	20
2.3.4 Voxel sampling . . . . .	21
2.3.5 Consistency metrics . . . . .	23
2.3.6 Carving as an optimisation problem . . . . .	24
2.3.7 Probabilistic carving . . . . .	26
2.3.8 Photo-consistency on graphics hardware . . . . .	29
2.4 Summary . . . . .	30
<b>Chapter III: Parameterised Photo-Consistent Geometry</b>	<b>33</b>
3.1 Overview . . . . .	33
3.2 Fitting photo-consistent geometry . . . . .	35
3.2.1 The surface model . . . . .	37
3.2.2 Hierarchical transformations . . . . .	38

3.2.3	The rendering equation . . . . .	41
3.3	The re-projection likelihood . . . . .	43
3.3.1	Estimating the surface texture . . . . .	44
3.4	Surface consistency on graphics hardware . . . . .	46
3.4.1	Adapting the likelihood . . . . .	48
<b>Chapter IV:</b>	<b>Image-Space Photo-Consistency</b>	<b>49</b>
4.1	Stereo photo-consistency . . . . .	49
4.2	Image mapping via the hypothesised surface . . . . .	51
4.2.1	Image mapping with occlusion . . . . .	52
4.2.2	Generating synthetic images . . . . .	53
4.3	Graphics implementation . . . . .	54
4.3.1	Stereo occlusion . . . . .	54
4.3.2	Configuring the graphics pipe . . . . .	56
4.4	Shape inference with photo consistency . . . . .	59
4.5	Experiments . . . . .	62
4.5.1	Noise testing . . . . .	65
4.5.2	University poster sequence . . . . .	68
4.5.3	Cardboard box sequence . . . . .	73
	Optimising camera parameters . . . . .	76
4.6	Limitations of image-space occlusion . . . . .	78
4.6.1	Image-space visibility constraints . . . . .	79
4.6.2	Toward a surface-space occlusion likelihood . . . . .	81
<b>Chapter V:</b>	<b>Towards a Surface Occlusion Metric</b>	<b>83</b>
5.1	Surface-based visibility constraints . . . . .	84
5.2	Generating view-specific textures . . . . .	85
5.3	Generating surface textures with graphics hardware . . . . .	88
5.4	Fixed-pipe back-projection . . . . .	91
5.4.1	Pre-processing triangles . . . . .	93
5.4.2	Combining colour and occlusion . . . . .	95
5.5	Hardware texture algorithm . . . . .	96
5.5.1	The vertex shader . . . . .	97
5.6	Results . . . . .	99

<b>Chapter VI:</b>	<b>Assembling The Composite Texture</b>	<b>103</b>
6.1	Composition on graphics hardware . . . . .	103
6.2	Composition in clamped colour buffers . . . . .	104
6.2.1	Computing surface visibility . . . . .	105
6.2.2	Compositing view-specific textures . . . . .	106
	Storing visibility weights in the frame-buffer . . . . .	108
	Modulating visibility weights in the fragment processor . . . . .	108
6.2.3	Summary . . . . .	109
6.3	Variable-weight composition . . . . .	112
6.3.1	Compositing variable weight view-specific textures . . . . .	114
6.3.2	Weighted composite texture experiments . . . . .	115
<b>Chapter VII:</b>	<b>Surface-Space Photo-Consistency</b>	<b>117</b>
7.1	Surface consistency likelihood . . . . .	117
7.1.1	The occlusion prior . . . . .	119
7.1.2	Identifying interior points . . . . .	120
7.1.3	Intersection in graphics hardware . . . . .	122
7.1.4	Computing the surface-space likelihood . . . . .	125
7.2	Experiments . . . . .	129
7.2.1	Synthetic stair test . . . . .	132
7.2.2	Cardboard box test revisited . . . . .	135
7.2.3	Template test . . . . .	140
7.2.4	House test . . . . .	142
7.3	Gradient-based optimisation . . . . .	146
7.4	Visibility bias . . . . .	149
<b>Chapter VIII:</b>	<b>Conclusion</b>	<b>151</b>
8.1	Future Work . . . . .	156
8.1.1	Extended primitives . . . . .	156
	Parameterised surfaces . . . . .	156
	Generalised cylinders . . . . .	156
	Extending constructive solid geometry . . . . .	158
8.1.2	Reconstruction without prior calibration . . . . .	158
8.1.3	Scene construction . . . . .	159
8.1.4	Estimating specular surfaces . . . . .	161

<b>Appendix A:</b>	<b>Profiling</b>	<b>163</b>
A.1	Generating view-specific textures . . . . .	163
A.1.1	Triangle scalability . . . . .	164
A.1.2	Back-facing culling . . . . .	165
A.1.3	Image-plane clipping . . . . .	165
A.1.4	Conclusion . . . . .	166
A.2	Hypothesis evaluation rate . . . . .	168
<b>Appendix B:</b>	<b>The Graphics Pipeline</b>	<b>171</b>
B.1	OpenGL . . . . .	171
B.1.1	The vertex processor . . . . .	173
B.1.2	The fragment processor . . . . .	174
B.1.3	Frame-buffer operations . . . . .	175
B.1.4	Summary . . . . .	176
B.2	Camera model . . . . .	178
<b>Bibliography</b>		<b>181</b>

## LIST OF FIGURES

2.1	Line-based re-projection error is given by the distance between the marked, reference edge and the hypothesised projection. . . . .	9
2.2	The visual hull is constructed by intersecting the silhouette cone from each reference image. . . . .	10
2.3	Photo-consistent points have a similar colour in all views where they are visible. . . . .	14
2.4	Points on the reference surface will appear inconsistent unless occlusion is used to decide which cameras can vote for a point's consistency. . . . .	15
2.5	Photo-consistency is a property held by an infinite number of hypothesised surfaces. . . . .	16
2.6	Two classes of depth-compatible scene configurations. . . . .	17
2.7	Arbitrary configurations can break the ordinality constraint. . . . .	18
2.8	Space carving sweeps along each scene axis. . . . .	18
2.9	Partitioning cameras can include inconsistent voxels. . . . .	19
2.10	Removing voxels can reveal voxels that are not adjacent. . . . .	19
2.11	Various scene configurations and the envelopes used to initialise space carving. . . . .	20
2.12	Camera calibration errors mean surface voxels can appear inconsistent while voxels not on the surface appear consistent. . . . .	23
3.1	The surface can be radically transformed by changing only a few parameters, each with their own prior distributions. . . . .	34
3.2	Photo-consistency with geometry constraints can be used to reduce surface ambiguity rather than finding only the largest consistent shape. . . . .	35
3.3	Examples of scene-graph primitives. . . . .	38
3.4	Scene graph construction for a house model. . . . .	39
3.5	Binding scene-graph parameters to transformations and the transformations to links between scene-graph nodes. . . . .	39
3.6	Projectively mapping texture to image-space by finding the nearest surface point to a camera and mapping it to its corresponding texture image. . . . .	43
3.7	The photo-consistency constraint. . . . .	45



3.8	Because corresponding scene points are unlikely to align in image-space, they must be mapped to a common space for comparison. . . . .	47
4.1	The variance can be approximated by pairwise comparison in all views.	50
4.2	The colour under a point's projection in the source image is mapped to its projection in the target image if it is visible in both images. . . . .	51
4.3	An overview of the process to generate synthetic images of the hypothesised surface. . . . .	53
4.4	Shadow mapping used to test occlusion from a light-source. . . . .	55
4.5	Generating synthetic views of the surface at truth. . . . .	58
4.6	The juxtaposition of two synthetic views with its corresponding target reference image at truth. . . . .	58
4.7	The surface's sampling rate is driven by the target image's resolution. . .	59
4.8	The six synthetic and consistency images for the hypothesised parameter vector illustrated in Figure 4.9(b). . . . .	60
4.9	The accumulated consistency images for four hypothesised vectors. . . .	60
4.10	The three synthetic reference images used for testing the image-space likelihood. . . . .	62
4.11	The image-space likelihood when varying a subset of the scene's transformation parameters while the others are kept at truth. . . . .	63
4.12	The image-space likelihood when varying the camera parameters while the surface is kept at truth. . . . .	64
4.13	An illustration of the surface priors generated by overlaying random samples. . . . .	65
4.14	The effect of additive Gaussian noise on an 24-bit RGB image with $\mu = 0$ and $\sigma$ at 0, 16 and 32. . . . .	67
4.15	The average relative scene parameter variance for increasing levels of image noise. . . . .	67
4.16	The reference images of the the poster cylinder sequence. . . . .	68
4.17	The translation and scale priors for the poster sequence. . . . .	69
4.18	Re-projection of the university poster sequence using one of the reference cameras. . . . .	70
4.19	Back-projecting the reference images onto the reconstructed surface . .	71
4.20	Novel views of the poster-cylinder reconstruction (1). . . . .	72
4.21	Novel view of the poster-cylinder reconstruction (2). . . . .	73
4.22	The priors probabilities, the parameter values corresponding to the calibration cube, and the MAP estimate for the cardboard box test. . . .	74

4.23	Reference images for the cardboard cube sequence. . . . .	75
4.24	Re-projection of the MAP surface using the reference cameras. . . . .	75
4.25	Novel views of the cardboard box reconstruction. . . . .	75
4.26	The relative log likelihoods of the MAP estimate for different priors. . . .	76
4.27	The initial Tsai estimate and two MAP estimates with different priors for the camera extrinsics. . . . .	77
4.28	A surface point must be visible in multiple views in order to determine its photo-consistency. . . . .	79
4.29	Three classes of scene visibility ranging from total visibility, partial visibility and total occlusion. . . . .	80
4.30	A conditional visibility chain over three iterations. . . . .	81
5.1	A comparison of image-space and surface-based treatment of occlusion.	84
5.2	View-specific textures are generated by back-projecting the reference image onto the surface. . . . .	85
5.3	Photo-consistency is determined by constructing a set of view-specific textures and comparing them against the ‘composite’ texture. . . . .	86
5.4	Texture co-ordinates define a map from the surface’s geometry to an image representing the surface’s texture. . . . .	87
5.5	Mapping points in the reference image to the surface texture. . . . .	88
5.6	The fixed-pipe back-projection algorithm. . . . .	91
5.7	OpenGL’s texture semantics treat textures as continuous entities. . . . .	93
5.8	Triangles are clipped by scaling the depth of each vertex according to its corresponding depth in scene-space. . . . .	98
5.9	Plane texture generated by back-projecting one image onto the ground plane. . . . .	100
5.10	View-specific textures can recreate reference images. . . . .	100
5.11	Three view-specific plane textures stylised by shading occluded pixels. .	101
6.1	The surface’s visibility map is computed by modulating each view- specific visibility mask with the frame-buffer. . . . .	105
6.2	Per-pixel contribution weights are computed by inverting the visibility image. . . . .	106
6.3	Assembling view-specific textures together into a single composite texture representing the hypothesised surface reflectance. . . . .	107
6.4	Overview of assembling the composite texture. . . . .	109
6.5	An example of composite texture formed from nine view-specific textures.	110

6.6	The composite texture generated by varying surface parameters. . . . .	111
6.7	Filtering issues when sampling oblique surfaces. . . . .	113
6.8	Weight functions. . . . .	115
6.9	The juxtaposition of fixed and varying weight textures. . . . .	116
7.1	Differences in colour between corresponding texels indicates that the hypothesised surface is incorrect. . . . .	118
7.2	The surface model allows objects to be embedded to give the external appearance of more complicated surfaces. . . . .	119
7.3	Using CSG to find interior surfaces. . . . .	120
7.4	The edges of shadow volumes partition the space into lit and unlit regions. . . . .	121
7.5	The world space is mapped to texture space for intersection testing. . . . .	122
7.6	Using stencil buffers to compute surface intersection. . . . .	124
7.7	Photo-consistency from hypothesised surfaces reprojected into two views (1). . . . .	126
7.8	Photo-consistency from hypothesised surfaces reprojected into two views (2). . . . .	127
7.9	Nine synthetic cube/plane reference images. . . . .	130
7.10	Synthetic optical centre likelihood. . . . .	130
7.11	Synthetic focal length likelihood. . . . .	130
7.12	Synthetic scale likelihood. . . . .	131
7.13	Synthetic translation likelihood. . . . .	131
7.14	Synthetic rotation likelihood. . . . .	131
7.15	The six synthetic reference images for the stairs test. . . . .	133
7.16	The composite textures for the stair test. . . . .	133
7.17	An annealed MCMC random walk visualisation for the stairs test. . . . .	134
7.18	Three stair reconstructions. . . . .	134
7.19	Synthetic views of the stair test reconstruction. . . . .	134
7.20	Reference images for the second cardboard box test. . . . .	135
7.21	The re-projection of the MAP estimates from the three different cardboard box tests. . . . .	136
7.22	The luminance histogram of a corresponding $128 \times 128$ pixel region from two views. . . . .	137
7.23	Micro-facet surfaces may <i>appear</i> to be specular due to occlusion. . . . .	137
7.24	A juxtaposition of fixed- and variable-weight composite textures. . . . .	137
7.25	The composite texture and consistency images. . . . .	139

7.26	Three reference images of the house-template test. . . . .	141
7.27	The re-projection of the template-house test using the reference camera parameters. . . . .	141
7.28	Synthetic images of the template-house test generated from novel view-points using the maximum <i>a posteriori</i> composite texture. . . . .	141
7.29	The seven reference images and prior distribution for the house reconstruction test. . . . .	143
7.30	Synthesised images of the house reconstruction from novel view-points.	143
7.31	The juxtaposition of the two house solutions. . . . .	144
7.32	A novel view of the house reconstruction. . . . .	144
7.33	The juxtaposition of synthetic images from two optimisations of the house test. . . . .	145
7.34	Change in surface space involves a corresponding change in image space.	147
7.35	Image plateaus lead to ambiguous likelihoods. . . . .	147
7.36	Approximating images by discrete pixel sampling. . . . .	148
7.37	The $\chi^2$ PDF . . . . .	150
7.38	Sample size versus probability. . . . .	150
8.1	Photo-consistency could also be measured over parameterised surfaces.	157
8.2	Examples of generalised cylinders. . . . .	157
8.3	An example of constructive solid geometry. . . . .	157
8.4	Scene construction example. . . . .	160
8.5	Specular surfaces cannot be estimated by a likelihood function that assumes diffuse reflectance. . . . .	161
A.1	The scenes designed to test the fixed- and programmable-pipe back-projection algorithms. . . . .	166
A.2	The texture rate with respect to an increasing number of triangles. . . .	167
A.3	The texture rate with respect to an increasing proportion of back-facing triangles. . . . .	167
A.4	The texture rate with respect to incrementally clipped triangle stack. . .	167
A.5	The configuration for evaluating the texture rate with respect to number of cameras and triangles. . . . .	168
A.6	The texture rates of the surface-based and image-based likelihoods. . .	170
B.1	The OpenGL pipe. . . . .	172
B.2	The OpenGL vertex processor. . . . .	173

B.3	The OpenGL fragment+frame-buffer engine. . . . .	174
B.4	The pin-hole camera model. . . . .	178

## LIST OF ALGORITHMS

1	Warping a single image into a new synthetic view . . . . .	57
2	Fixed-pipe texture algorithm. . . . .	96
3	Programmable-pipe vertex shader. . . . .	97
4	Compositing view-specific textures. . . . .	112
5	Identifying interior texels. . . . .	125

## ACKNOWLEDGEMENTS

I would like to thank my supervisors Mike Brooks and Anton van den Hengel, and to Anthony Dick who offered advice and encouragement. My thanks also goes to Thorsten Thormählen, whose help with calibration was very much appreciated, and Darren Gawley for questioning the existence of zero-dimensions. I would also like to thank Rhys Hill, Daniel Pooley and Travis Olds for many interesting discussions and shared experiences.

This would not have been possible without the support and understanding of my family, and I thank them for their encouragement, empathy and quiet patience.

## NOTATION

### PROBABILITY

---

$\Pr(X)$	the probability of event $X$
$\Pr(X   Y)$	the conditional probability of event $X$ given $Y$
$\mathcal{L}(X   Y)$	the likelihood that $X$ given $Y$
$\mathcal{U}(a, b)$	uniform distribution in the range $[a, b]$
$\mathcal{N}(\mu, \sigma)$	normal distribution with mean $\mu$ and standard deviation $\sigma$

### IMAGES

---

$x(\mathbf{p})$	the colour of the point $\mathbf{p}$ in the image $x$
$\ddot{x}$	the RGB channels of image $x$
$\dot{x}$	the $\alpha$ -channel of image $x$
$r_i$	the $i^{\text{th}}$ reference image
$\mathcal{I}$	the set of reference images, $\mathcal{I} = \{r_i\}$
$s_{s,t}$	the synthetic image of the hypothesised surface rendered by camera $t$ while using $r_s$ as the surface texture.
$v_i$	the view-specific texture associated with $r_i$
$t$	the estimated surface texture
$\mathcal{C}$	the set of surface textures, $\mathcal{C} = \{t_i\}$

### GRAPHICS PIPE

---

$P_G$	OpenGL pipe-line's projection matrix
$M_G$	OpenGL pipe-line's modelview matrix
$T_G$	OpenGL pipe-line's texture matrix



## GEOMETRY

---

$\underline{p}$	Euclidean 3D points and vectors
$M$	matrix
$M_n$	row $n$ of matrix $M$
$\mathcal{P}$	a geometric primitive
$\mathcal{S}$	the scene-graph
$\rho$	scene-graph parameter
$\Psi$	the scene-graph parameter vector, $\Psi = \{\rho_i\}$
$\mathcal{S}(\Psi)$	an instance of the scene-graph parameterised by the vector $\Psi$
$t$	triangle, $t = \{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3\}$

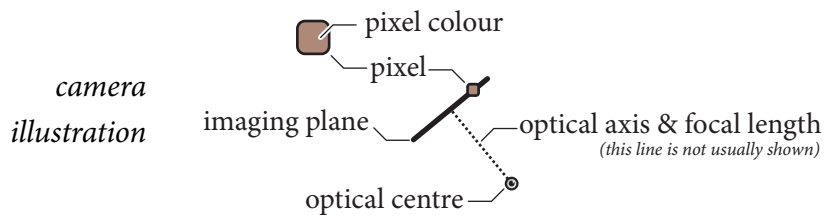
## PROJECTIVE GEOMETRY

---

$p$	homogeneous 3D point
$T_{xyz}$	$4 \times 4$ translation matrix; subscript indicates free axes
$R_{xyz}$	$4 \times 4$ rotation matrix; the subscript indicates free axes
$S_{xyz}$	$4 \times 4$ scale matrix; the subscript indicates free axes
$K$	intrinsic matrix
$E$	extrinsic matrix
$P$	$3 \times 4$ projection matrix to map 3D homogeneous points to 2D homogeneous points, $P = KE$
$\hat{K}$	$4 \times 4$ projection matrix to map 3D homogeneous points to eye co-ordinates
$\hat{P}$	graphics projection matrix, $\hat{P} = \hat{K}E$

## ILLUSTRATION

---



“The time has come,” the Walrus said,  
“To talk of many things.”



Lewis Carroll  
*Through the Looking-Glass*