Visual Contact Estimation

Christopher. D. McCarthy

July, 2010

A thesis submitted for the degree of
Doctor of Philosophy
at the Australian National University

Department of Information Engineering
College of Engineering and Computer Science
Australian National University

Declaration

This is to certify that this thesis comprises only my original work except where indicated in the preface; due acknowledgment has been made in the text to all other material used; the thesis is approximately 75,000 words in length, inclusive of footnotes, but exclusive of tables, maps, appendices and bibliography.

Acknowledgments

Firstly, I would like to thank my supervisor, Dr. Nick Barnes. I particularly thank him for his dedication and support throughout my PhD candidature, his insightful advice, and his valuable assistance in putting this thesis together . It has been a privilege to work with him. Thanks also to the other members of my PhD advisory committee, Professor Mandyam Srinivasan and Professor Giulio Sandini, for their valuable input on aspects of the work presented in this thesis.

In 2007, I was lucky enough to spend 6 months visiting the Italian Institute of Technology’s (IIT) Robotics, Brain and Cognitive Sciences laboratory in Genova. I thank Professor Sandini and the IIT for hosting my stay. I also thank Dr. Giorgio Metta for his guidance and assistance, and Ingrid Sica for her efforts in arranging my Italian visa, and my accommodation during that time. The six months spent at IIT was without doubt the highlight of my PhD.

Many thanks to Luke Cole for his efforts in building and maintaining the mobile robot platform used for results presented in Chapters 6 and 7 of this thesis. Thanks also to The University of Melbourne’s Department of Computer Science and Software Engineering for providing the robot used in onboard trials described in Chapter 5.

The work presented in Chapter 9 of this thesis forms part of a larger collaborative study titled ’*Hazard perception and cognitive ageing in older drivers: investigation and development*’. I thank Professor Kaarin Anstey, and Chris Hatherly at the Centre for Mental Health Research, Australian National University, and Dr. Mark Horswill, School of Psychology, University of Queensland, for access to the hazard perception test footage. The footage used in this thesis was captured by Chris Hatherly.

Thank you to the Australian National University, and NICTA, for providing me with the necessary financial support to complete this PhD. Thanks also to the staff and students of the Department of Information Engineering, and the administration staff within the College of Engineering and Computer Science for their assistance throughout

my candidature. In particular, I would like to thank Professor Rod Kennedy as head of the Department of Information Engineering, and the administration staff at RSISE (at various times), Rosemary Shepherd, Lesley Goldburg, Julie Arnold, Elspeth Davies, Debbie Pioch and Marie Katselas. Thanks also to the administration staff at NICTA CRL. In particular, Steve Marlor and Kirk Hellyar.

Many thanks to everyone in the Computer Vision group (VISTA) within NICTA's Canberra Research Lab for providing such an inspiring, motivating and friendly working environment. Being amongst such quality researchers is both inspiring and intimidating, but always educational. A special thank you to my fellow *level 1* office mates: Nathan Brewer, Peter Carr, Gary Overett, Tim Raupach, and Tamir Yedidya, for providing such a friendly and relaxed environment to work.

Throughout my PhD, I have also had the good fortune to meet many new people, and get involved in fun and rewarding extra curricular activities. I would like to make special mention of the ANU Postgraduate and Research Students Association (PARSA), and the fantastic people I got to work with during my time there.

I would also like to take this opportunity to thank my parents, Mike and Shirley McCarthy. My ability to pursue a PhD, let alone complete one, is in no small part due to their tireless efforts in providing me with the love, support and necessary opportunities to learn and develop the skills and self discipline required to get this done. I have never been more appreciative (or aware of the importance) of these lessons until now.

And finally, a special thank you to my young little family. To my (nearly) nine month old son, Max, I say thank you for giving me a whole new reason to finish this thesis, and for being the fascinating boy you are. I look forward to spending more time with you, and less time being distracted by less important things. To my beautiful wife Affrica. After 250+ pages of thesis writing, it is difficult to find words to express how much your love and support has meant to me. Your compassion, your compromises, your patience, your humour, your devoted proof reading, and your unwaivering belief in me, are just some of the reasons why I love you, and why this thesis made it to submission. What can I say? We did it again!

Chris McCarthy.

Abstract

A fundamental capability of any navigation system is the perception of potential contact with surfaces in the environment. The efficiency and robustness of natural vision has motivated the development of biologically-inspired approaches to achieve this. Biological studies have highlighted the importance of visual motion (as perceived via optical flow) in the guidance of animal action. However, the use of optical flow in robot navigation systems remains problematic, impeded by measurement noise, environmental assumptions, and real-time constraints. This thesis proposes new biologically-inspired visual cues and algorithms for robust visual control and contact estimation from optical flow. We consider this primarily in the context of robot navigation and control.

We present a robust strategy for docking a mobile robot with near-frontal surfaces using optical flow divergence. Results show improved robustness during egomotion, allowing closer than previously reported stopping distances. We present a strategy for performing controlled approaches towards surfaces of arbitrary orientation, providing the first unified control law for landing and docking. Velocity and heading control is achieved using only the maximum flow divergence on the view sphere. We present an insect-inspired structure-from-motion scheme using spherical optical flow from a hemispherical fish-eye sensor, providing the first demonstration of real-time depth map recovery from dense optical flow estimation. In dynamic environments, we investigate the use of optical flow to predict the time and location of impact of incoming objects. We consider this in the context of a stationary camera, as well as for on-road driver hazard perception assistance.

We conclude that robustness in flow-based control schemes can be improved if system dynamics are handled in the image domain. This can be achieved by prioritising visual cues conveying a relationship between self-motion and scene structure over explicit structure-from-motion recovery in the control loop. Results suggest a wide-angle spherical projection model is well-suited for visual contact estimation from optical flow.

Publications

Several contributions presented in this thesis have been published elsewhere by the author. We list these below:

Journal

- Chris McCarthy, Nick Barnes and Rob Mahony, 'A robust docking strategy for a mobile robot using flow field divergence.' *IEEE Transaction on Robotics*, Volume 24(4), 2008.

Conference

- Chris McCarthy, Nick Barnes and Mandyam Srinivasan, 'Real time biologically-inspired depth maps from spherical flow.' In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA 2007)*, Rome, Italy, 2007.

- John Lim, Chris McCarthy, David Shaw, Nick Barnes and Luke Cole, 'Insect Inspired Robots.' In *Proceedings of the 2006 Australiasian Conference on Robotics and Automation (ACRA 2006)*, Auckland, New Zealand, 2006.

- Chris McCarthy and Nick Barnes, 'A robust docking strategy for a mobile robot using flow field divergence.' In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2006)*, Beijing, China, 2006.

Workshop

- Chris McCarthy, Nick Barnes, Kaarin Anstey and Mark Horswill, 'Towards a Hazard Perception Assistance System using Visual Motion.' In *Proceedings of the ECCV Workshop on Computer Vision Applications for the Visually Impaired (CVAVI 08)*. Marseille, France, 2008.

Contents

1 Introduction 1

1.1 Motivation . . . 1

1.1.1 Exploring visual perception through robotics . . . 2

1.1.1.1 The challenge of computational visual perception . . . 3

1.1.2 An ecological approach to visual navigation . . . 3

1.1.2.1 Visual motion . . . 4

1.1.3 Visual contact estimation for navigation and perception . . . 4

1.1.3.1 Collision avoidance . . . 6

1.1.3.2 Docking/Landing . . . 6

1.1.3.3 Threat avoidance in dynamic environments . . . 6

1.1.4 Contact estimation from optical flow . . . 6

1.1.4.1 Time-to-contact . . . 7

1.1.4.2 Time-to-contact estimation for robot navigation . . . 7

1.1.5 Systems of visual control using visual motion . . . 7

1.1.6 Projective geometries for visual contact estimation . . . 8

1.2 Core contributions of this thesis . . . 9

1.3 Thesis Overview . . . 10

2 Motivation and Approach 13

2.1 Introduction . . . 13

2.2 Perception for robot navigation . . . 14

2.2.1 The sense-plan-act paradigm . . . 14

2.2.2 Behaviour-based navigation . . . 15

2.2.3 Visual perception under a behaviour-based framework . . . 16

2.3 Classical computer vision for navigation and perception . . . 16

2.3.1 Marr's theory of computational vision . . . 17

2.3.2 Traditional 3D scene reconstruction in computer vision 17
2.4 Vision-guided robot navigation 19
2.4.1 Visual navigation using global maps 19
2.4.1.1 Grid-based mapping 19
2.4.1.2 Simultaneous localisation and mapping (SLAM) 20
2.4.1.3 Summary of global map-based visual navigation 21
2.4.2 Visual navigation using local maps 22
2.4.2.1 Summary of local map-based navigation 23
2.4.3 Visual navigation without maps 23
2.4.3.1 Visual navigation in semi-structured environments . . . 23
2.4.3.2 Visual servoing and appearance-based navigation 24
2.4.3.3 Reactive vision-guided obstacle detection and avoidance 25
2.4.3.4 Summary of visual navigation without maps 26
2.5 Biological vision for robot navigation 27
2.5.1 Gibson's theory of direct perception 27
2.5.2 Active perception 27
2.5.3 Active perception with visual motion 28
2.6 Visual motion for robot navigation 30
2.6.1 Corridor following 30
2.6.2 Visual odometry 31
2.6.3 Altitude regulation, hovering and station keeping 31
2.6.4 Landing and docking 31
2.6.5 Discussion of current visual motion-based navigation 32
2.6.6 Measuring visual motion: optical flow versus point matching . . 32
2.7 Summary 33

3 Estimating and Interpreting Optical Flow 35
3.1 Introduction 35
3.2 Optical flow 36
3.2.1 Derivation of optical flow 36
3.2.1.1 Common definitions 36

3.2.1.2 Optical flow under spherical projection 37
3.2.1.3 Optical flow under perspective projection 38
3.2.2 Estimating the optical flow field 39
3.2.2.1 The aperture problem 39
3.2.2.2 Classes of optical flow techniques 41
3.2.2.3 Gradient-based optical flow estimation 42
3.2.2.4 Advantages of local gradient-based flow estimation . . . 44
3.2.2.5 Enhancements and extensions 45
3.2.3 Summary of optical flow estimation 46
3.3 Inferring local structure and motion from optical flow 46
3.3.1 Differential versus discrete motion 47
3.3.2 Differential invariants of visual motion 47
3.3.2.1 Optical flow across a rigid surface 47
3.3.2.2 Local flow field structure and decomposition 48
3.3.2.3 Relating invariants to motion and structure 50
3.3.2.4 Interpreting differential invariants 51
3.3.2.5 Coupling of deformation with divergence and curl . . . 53
3.3.3 Closed-form solutions to local structure and motion 53
3.3.4 The planar surface ambiguity 54
3.3.4.1 Unique solutions for the planar case 56
3.3.4.2 Planar structure-from-motion under general motion . . 56
3.3.5 Limitations of closed-form structure-from-motion 57
3.4 Egomotion estimation from optical flow 58
3.4.1 Egomotion from global flow field samples 58
3.4.2 Egomotion from the focus of expansion 59
3.4.2.1 FOE estimation 59
3.4.2.2 Structure and egomotion from the FOE 60
3.5 Robust depth map recovery 61
3.6 Real-time structure-from-motion for navigation 62
3.6.1 Example systems 62
3.6.2 Summary of structure-from-motion based navigation 64

3.7 Issues for structure-from-motion recovery . . . 64
3.7.1 Flow field noise . . . 64
3.7.2 Translation-rotation ambiguity . . . 66
3.7.3 Qualitative versus quantitative structure from motion . . . 66
3.8 Insect-inspired structure-from-motion for navigation . . . 67
3.8.1 Geometric advantages of spherical projection . . . 67
3.8.2 Estimating egomotion from spherical optical flow . . . 68
3.8.3 Summary of insect-inspired structure-from-motion . . . 69
3.9 Summary . . . 69

4 Time-to-contact for visuo-motor control 73
4.1 Introduction . . . 73
4.2 Time-to-contact . . . 74
4.2.1 Definition and motivation . . . 74
4.2.2 Biological support . . . 75
4.3 Inferring time-to-contact from optical flow . . . 75
4.3.1 Generalising time-to-contact to any viewing direction . . . 76
4.4 Computing time-to-contact . . . 77
4.4.1 Time-to-contact from flow field models and approximations . . . 78
4.4.2 Time-to-contact from closed-contours . . . 79
4.4.2.1 Green's theorem . . . 79
4.4.2.2 Closed-contour examples . . . 80
4.4.3 Time-to-contact from space-variant maps/sensors . . . 82
4.5 Time-to-contact for collision avoidance . . . 83
4.5.1 Summary of collision avoidance systems . . . 84
4.6 Time-to-contact for docking and landing . . . 84
4.6.1 The docking problem . . . 85
4.6.2 Docking and landing systems . . . 85
4.7 Issues for time-to-contact based visuo-motor control . . . 87
4.7.1 Flow field approximations . . . 87
4.7.2 Robustness during egomotion . . . 88

4.7.3 Surface orientation and alignment 89

4.7.4 Summary of time-to-contact for visuo-motor control 90

4.8 Summary 91

5 Robust Visual Docking using Flow Field Divergence 93

5.1 Introduction 93

5.2 Theory 94

5.2.1 Derivation of proposed time-to-contact estimator 94

5.2.2 Time-to-contact for a ground-based mobile robot 98

5.2.2.1 Constraints on rotation 98

5.2.2.2 Constraints on angle of approach (surface orientation) . 100

5.3 Experimental Results 100

5.3.1 Simulation experiment 101

5.3.2 Off-board time-to-contact experiments 101

5.3.2.1 Indoor image sequence 101

5.3.2.2 Outdoor image sequences 102

5.3.3 On-board docking experiment 103

5.3.4 Optical flow and FOE estimation 104

5.3.5 Results 105

5.3.5.1 Simulation results 105

5.3.5.2 Indoor image sequence results 106

5.3.5.3 Outdoor image sequence results 108

5.3.5.4 On-board docking results 108

5.4 Discussion 110

5.5 Summary 112

6 Unifying Docking and Landing using Spherical Flow Divergence 113

6.1 Introduction 113

6.2 Theory 114

6.2.1 Computing time-to-contact for planar surfaces 115

6.2.1.1 Advantages of a full view sphere 115

6.2.2 The point of maximum divergence 116

6.2.2.1 What does max-div represent? 120

6.3 Unifying landing and docking using max-div 120

6.3.1 Regulating approach velocity 120

6.3.2 Regulating approach angle 121

6.4 Open-loop performance evaluation 122

6.4.1 Omni-directional cameras employed 122

6.4.2 Indoor image sequences (controlled environment and motion) . . 124

6.4.3 Outdoor image sequences 125

6.4.3.1 Optical flow and divergence estimation 125

6.4.4 Open-loop results 126

6.4.4.1 Indoor landing sequence results 126

6.4.4.2 Outdoor landing sequence results 132

6.4.5 Open-loop results discussion 137

6.5 Closed-loop performance evaluation 138

6.5.1 Closed-loop simulation testing 138

6.5.2 Mobile robot landing/docking experiments 139

6.5.2.1 Implementation of max-div control scheme 140

6.5.3 Implementation of graze-landing model 141

6.5.4 Results 142

6.5.4.1 Closed-loop simulation results 142

6.5.4.2 Mobile robot landing/docking results 144

6.6 Discussion 150

6.6.1 Generality of max-div docking scheme 150

6.6.2 Robustness of time-to-contact estimation 152

6.6.3 Robustness of velocity control 153

6.6.4 Robustness of heading control (approach angle regulation) 153

6.6.5 Max-div model limitations and issues 154

6.6.5.1 Existence of a global divergence maximum 154

6.6.5.2 Divergence estimation 154

6.7 Implications for visuo-motor navigation 155

6.7.1 Global flow field invariants 155

6.7.2 FOE versus max-div for docking and landing 156
6.7.3 Visual input choices 156
6.8 Summary 157

7 Real Time Biologically-Inspired Depth Maps from Spherical Flow 159
7.1 Introduction 159
7.2 Estimating depth maps from spherical flow 160
7.2.1 Decomposition of flow about orthogonal great circles 160
7.2.2 Egomotion estimation 163
7.2.3 Egomotion estimation under planar motion 164
7.2.4 Generating relative depth maps 167
7.3 Performance evaluation 168
7.3.1 Simulation experiment 168
7.3.1.1 Method and implementation 168
7.3.1.2 Simulation results 169
7.3.2 Real-world experiments 169
7.3.2.1 Implementation of depth map recovery 169
7.3.2.2 On-board controlled environment experiment 170
7.3.2.3 Real-world navigation experiments 172
7.3.3 Results 175
7.3.3.1 On-board controlled environment results 175
7.3.3.2 Onboard corridor navigation results 177
7.3.3.3 Cluttered environment fly-through results 182
7.4 Discussion 183
7.5 Implications for general flow-based navigation 185
7.5.1 Structure-from-motion versus visuo-motor control 185
7.5.2 Towards systems of visual control 185
7.6 Summary 186

8 Time and location of impact prediction based on primate vision 189
8.1 Introduction 189
8.2 Background and motivation 190

8.2.1 Previous work 191

8.3 Proposed method 192

8.3.1 Computing angle of approach 194

8.3.2 Estimating location of impact 194

8.3.2.1 Direction of impact 194

8.3.2.2 Camera-centred location of impact 195

8.3.3 Estimating time-to-impact with the plane 197

8.4 Performance assessment 197

8.4.1 Implementation 197

8.4.2 Initial controlled-conditions testing 198

8.4.3 Live impact prediction test 202

8.5 Discussion 203

8.5.1 Future work 206

8.6 Summary 207

9 On-road hazard detection for driver assistance 209

9.1 Introduction 209

9.2 Background and motivation 210

9.2.1 Previous work 211

9.3 Proposed method 212

9.3.1 Identifying side-entering hazards with a stationary case 213

9.3.2 Identifying side-entering hazards during forward translation . . . 214

9.4 Implementation of side-entering detector 217

9.4.1 Generating the self-motion template 217

9.4.2 Hazard detection 217

9.4.3 The Hazard Perception Test 218

9.5 Performance assessment 219

9.5.1 Testing procedure 219

9.5.2 Results 219

9.6 Discussion 222

9.6.1 Future work 223

9.7 Summary 223

10 Conclusion and Future Work **225**

10.1 Key insights for visual contact estimation 226

10.2 Core contributions of this thesis 227

10.3 Further work 228

Bibliography **233**

List of Figures

1.1 Example visual motion fields resulting from camera motion. 5

2.1 Marr's pipeline for vision as an information-processing problem 17

2.2 The looming effect. 29

3.1 The aperture problem 40

3.2 The elementary fields of optical flow. 52

3.3 The planar ambiguity 55

3.4 A graphical depiction of the translation-rotation ambiguity 65

4.1 Green's theorem for the computation of divergence. 80

5.1 Geometric configuration 94

5.2 Sample frames used in off-board experiments. 102

5.3 Setup for on-board docking tests. 105

5.4 Simulation results for FOE-based time-to-contact 106

5.5 Off-board time-to-contact estimates. 107

5.6 Onboard docking results 109

6.1 Geometric framework for max div proof 117

6.2 Example of max-div property on the view sphere 119

6.3 Geometric model for approach angle control on view sphere 121

6.4 Sample frame from indoor landing sequence. 123

6.5 Sample frames from outdoor landing sequences. 124

6.6 Sample frames from indoor landing sequence for 0^o approach 127

6.7 Sample frames from indoor landing sequence for 22.5^o approach. 128

6.8 Sample frames from indoor landing sequence for 45^o approach. 129

6.9 Sample frames from indoor landing sequence for 67.5^o approach. 130

6.10 Time-to-contact results for open-loop, indoor landing sequences. 131

6.11 Sample frames from outdoor grass-landing sequence (straight). 133

6.12 Sample frames from outdoor grass-landing sequence (rotating). 134

6.13 Sample frames from outdoor cement landing sequence. 135

6.14 Outdoor open-loop time-to-contact results 136

6.15 Side view of on-board experimental workspace. 139

6.16 View from onboard camera of closed-loop experiments 140

6.17 Graze-landing model velocity curves. 142

6.18 Velocity-time profiles for onboard trials. 143

6.19 Trajectory profiles for onboard trials. 143

6.20 Sample overhead images from onboard experiments. 144

6.21 Sample output images from max-div onboard trials: 0^o. 145

6.22 Sample output images from max-div onboard trials: 67.5^o. 146

6.23 Plotted paths of robot from max-div onboard trials. 147

6.24 Histograms of relative frequency of max-div image locations. 148

6.25 Velocity-time profiles recorded during all on-board trials. 149

6.26 Time-to-contact values recorded during all on-board trials. 151

7.1 Optical flow on the view sphere. 161

7.2 Geometric framework for de-rotation on the view sphere. 162

7.3 Flow vectors follow great circles passing through the FOE and FOC . . 164

7.4 Nelson and Aloimonos egomotion algorithm 165

7.5 Adaptation of Nelson and Aloimonos de-rotation for planar motion. . . 166

7.6 Images of workspace for controlled environment experiment 171

7.7 Robot position/orientation during controlled environment experiment. . 173

7.8 Robot motion data recorded during controlled environment experiment . 174

7.9 Sample frames from the corridor navigation sequence 175

7.10 Sample frames from the hand-held cluttered fly-through sequence 175

7.11 Egomotion estimation from controlled environment sequence. 176

7.12 Sample output from controlled environment experiment. 178

7.13 Sample depth maps from controlled environment experiment. 179

7.14 Sample depth maps from onboard camera during corridor navigation. . 180
7.15 Sample frames and depth maps from kitchen fly-through. 181

8.1 Geometry of impact prediction on a planar body 195
8.2 Estimation of impact distance on a planar body 196
8.3 Sample frames from landing sequence experiments. 200
8.4 Estimated angles of approach for landing sequences. 201
8.5 Estimated location of impact for both landing sequences. 201
8.6 Ratio of object time-to-contact along motion and viewing direction. . . 202
8.7 Sample output frames from live test: open hand 204
8.8 Sample output frames from live test: closed fist 205

9.1 Geometric framework for side-entering hazard detection. 212
9.2 Sample side-entering hazard scenario. 218
9.3 Sample hazard detections from video segment testing. 221

List of Tables

3.1 Summary of Structure-from-motion (SFM) techniques discussed in this chapter. 70

4.1 Summary of time-to-contact estimation scheme assumptions. 90

4.2 Summary of time-to-contact schemes in the control loop 91

6.1 Sphere versus plane for time-to-contact estimation. 116

6.2 Results for indoor landing sequences 126

6.3 Approach angle data for straight outdoor landing sequences. 137

6.4 Performance statistics for closed-loop max-div trials. 145

7.1 Simulation error measures 168

9.1 Hazard perception test results using the side-entering hazard detector. . 220

9.2 Detection results broken into side-entering hazard classes. 221

Chapter 1

Introduction

1.1 Motivation

So seamlessly does vision serve the needs of animal activity, its complexity and resilience is easily taken for granted. Yet, it has been estimated that at least forty independently evolved eye designs exist in the natural world [Dawkins 1996]. Each design differs in size, field of view, resolution, configuration and geometry. Underlying these physical designs are visual processes charged with the task of extracting meaningful information from the received light, supporting capabilities such as object recognition, and navigation. Ecological studies suggest these visual processes are highly tuned to an animal's environment and the needs of survival within that environment [Gibson 1950; Gibson 1979]. Vision therefore serves a multitude of purposes, defined by the tasks it supports, and the conditions in which it operates.

Determining the role of vision in the guidance of action has been a focus of science and philosophical discussion for many years. Biologists, psychologists, psychophysiscists, neuroscientists, among others, have sought to better understand the complex interplay between visual perception and the behaviour of animals (including humans). More recently, these researchers have been joined by those who seek to build artificial vision systems. In particular, the field of computer vision has sought to develop algorithms capable of interpreting digital imagery to infer scene structure and self-motion. One aim of this research is to provide vision algorithms capable of supporting autonomous vehicle guidance in unknown environments. Such techniques may also be embedded in technologies to improve or enhance human perception and mobility, particularly for the visually impaired.

1.1.1 Exploring visual perception through robotics

The complexity of vision is exemplified in its application to robot navigation. The concept of a robot navigating autonomously within an unknown environment has been a driving goal of many researchers for over forty years. This interest has been primarily motivated by a desire to increase the efficiency of human activity, and to replace humans in dangerous environments. The earliest areas of impact for such systems have been in industrial environments such as on automotive production lines, typically fixed in position and operating under precise, highly controlled conditions. More recently, non-fixed mobile robotic systems have been applied under less controlled environments, often inaccessible to humans. Examples include space exploration [Matthies et al. 2007], deep ocean [Kunz et al. 2008], subterranean [Roberts et al. 2003] and glacial surveying [Williams and Howard 2008], as well as search and rescue operations in disaster scenes [Birk and Carpin 2006].

The use of vision as a primary sensor for robot navigation offers several advantages over other sensor choices. In general, digital cameras are cheaper than other commonly used sensors such as laser range finders, sonar and radar. In addition, the ever decreasing size and weight of modern cameras allows for their easy integration and mounting on robotic platforms. This is particularly important where physical constraints exist, such as on aerial vehicles. Vision also offers a flexibility of use not present in other sensor choices. Information about colour, texture, and motion may be combined to perceive scene structure, surface shape and self motion. Moreover, this data can be communicated at high speed and made available through a single input source. Other sensors typically provide only single, fixed types of data, requiring multiple streams of sensory inputs to be sampled.

Applying vision to robot navigation is also motivated by a broader interest in biological vision. The building and testing of biologically-inspired vision algorithms provides a useful platform for examining theories and computational models of vision in the context of real navigation tasks. This enables researchers to consider vision at a system-level, potentially uncovering limitations and/or implicit assumptions in theoretical models of biological vision systems. Such outcomes also serve to inform the building of more robust vision systems to assist human activity.

1.1.1.1 The challenge of computational visual perception

Despite considerable attention, the application of vision to robot navigation remains problematic. A primary issue is scaling vision algorithms to environments beyond for which they are designed. Issues such as lighting variation, image signal noise, and visual ambiguities (*e.g.,* depth versus size, shadows, reflections) all pose significant challenges for any vision-guided navigation system seeking to operate in the real world. In many cases, robust solutions currently do not exist. Overcoming such issues typically requires significant computational resources, potentially introducing considerable latency in the control loop. To alleviate these issues, assumptions about robot motion and/or environmental conditions are often required, thereby reducing system scalability.

In applying computer vision algorithms to robot navigation, the classical role of vision has been to facilitate the construction and maintenance of some internal geometric world model. Based on this, robot motor actions can be formulated, and if needed, adjusted as conditions change. However, the real-time recovery and maintenance of such models in complex, real-world environments has proven to be a difficult and challenging problem. While algorithmic and hardware advances continue to alleviate such concerns, these inherent difficulties have motivated researchers to consider alternative approaches to visual navigation.

1.1.2 An ecological approach to visual navigation

From a broad study of animals in their environment, Gibson [1979] argued that the world itself provides its own best model, implying that no internal representation of the environment is required to support navigation. Gibson argued that all environmental structure and self-motion information is made directly available to animals through the perception of motion. Movement may therefore be controlled directly from visual information expressed through invariants present in the constantly changing image. Subsequent neuroscience and psychophysical research suggests most biological vision systems are equipped with specific neural mechanisms designed for processing and interpreting motion [Lee 1980]. The first stage of motion perception is measuring the visual motion.

1.1.2.1 Visual motion

Visual motion represents the apparent motion induced by the movement of surfaces in a scene with respect to the retina (or camera). It may be induced by the independent motion of objects in the scene, or by the motion of the observer with respect to the scene. Computationally, visual motion is represented as a 2D vector field in the image space, where vectors describe the movement of scene points in the image. These vector fields may be generated via the explicit correlation of features between time separated images (*point-matching*), or via the estimation of instantaneous image velocities (*optical flow*). Figure 1.1 shows some example visual motion fields.

In primate vision, visual motion is perceived in the early stages of the visual processing pathway [Duffy and Wurtz 1997]. Being immediately available and requiring minimal cognition to compute, it provides crucial support for low-level navigation and threat avoidance tasks [Fogassi et al. 1996; Zako et al. 2009]. Insects, with their immobile, fixed-focus eyes and low interocular separation rely almost exclusively on visual motion to infer range information [Srinivasan and Zhang 2004]. Estimating depth from visual motion is also simpler, computationally, than estimating depth from stereo, and thus better suited to the relatively simple nervous systems of insects. Insects have been shown to utilise visual motion to (i) navigate safely through narrow gaps, (ii) detect and avoid collisions with objects, (iii) distinguish objects from their immediate backgrounds, and (iv) orchestrate smooth landings [Srinivasan and Zhang 2004]. These ecological observations have inspired new approaches to the application of vision in robot navigation.

1.1.3 Visual contact estimation for navigation and perception

A fundamental capability of any visual navigation system is the perception of potential contact with surfaces in the scene. Most important is perceiving changes in their proximity resulting from either self-motion towards a surface, or the independent motion of

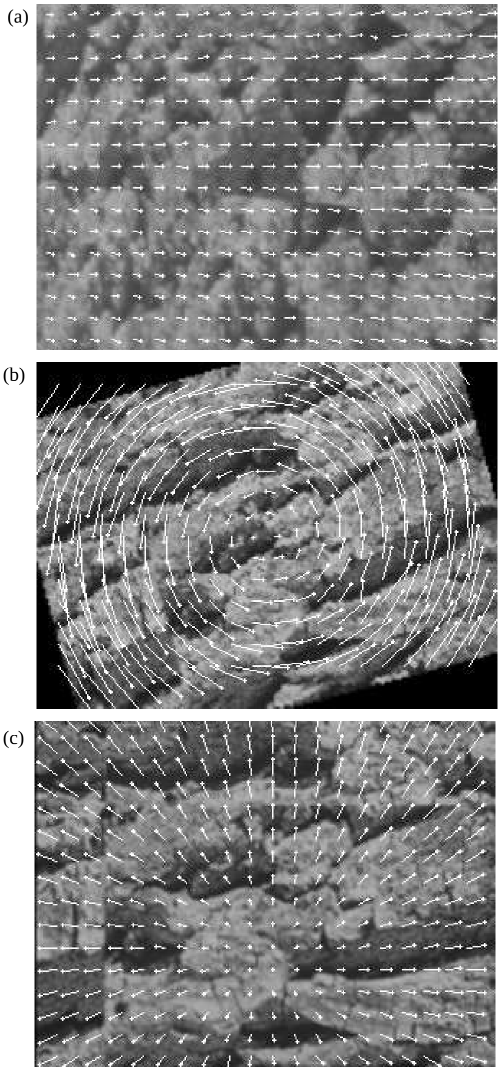

Figure 1.1: Example visual motion fields for (a) translational (b) rotational and (c) looming visual motion of the scene (a textured wall). Vectors show the apparent movement of image points due to the motion of the camera with respect to the surface.

objects towards the observer. Perceiving the relative change in distance between observer and environment is crucial to a number of important navigation tasks, including: collision avoidance, docking/landing and threat avoidance in dynamic environments.

1.1.3.1 Collision avoidance

The most basic visual navigation task when moving in an unknown environment is avoiding collision with objects in the scene. Given sufficient warning, a robot may alter its course, or halt further motion towards the surface.

1.1.3.2 Docking/Landing

The ability to perform controlled approaches to surfaces is an essential capability for any mobile robot seeking to interact with objects in its environment. Tasks such as plugging into a re-charging station, pallet lifting or transporting goods on a factory floor are common tasks requiring some form of docking manoeuvre to be performed. In flight, performing graze landings on run ways, or hover-down approaches are common tasks. Of particular importance in all these tasks is the controlled deceleration of the vehicle during the approach, such that velocity is zero (or close to zero in the case of graze landing) at the point of contact.

1.1.3.3 Threat avoidance in dynamic environments

An important capability of any system (artificial or biological) working in a dynamic environment is the ability to perceive independently moving objects. This is particularly important when the object poses an imminent threat of collision with the observer. To avoid collision, a looming object's rate of approach, and predicted trajectory with respect to the observer must be extracted.

1.1.4 Contact estimation from optical flow

All the necessary visual information to support the above applications is made directly available in the visual motion of the scene. The motion of an observer towards a surface, or analogously, a surface towards the observer, induces an apparent expansion, or *looming effect* in the observer's image (*e.g.,* Figure 1.1(c)). The rate of this expansion provides a direct means of estimating the immediacy of contact with the surface. This is commonly referred to as the *time-to-contact* or *time-to-impact*.

1.1.4.1 Time-to-contact

There is strong biological evidence suggesting time-to-contact is a commonly employed cue for detecting looming objects, and controlling motion towards surfaces. Srinivasan *et al.* [2000] observe how honeybees use visual motion to decelerate and perform smooth graze landings. Lee [1976] theorised that a human driver may visually control vehicle braking based on time-to-contact estimation obtained from image expansion. Primate and human studies have also shown that the looming effect causes defensive actions in response to perceived threats [Schiff et al. 1962; Bower and Broughton 1970].

1.1.4.2 Time-to-contact estimation for robot navigation

The direct availability of time-to-contact from visual motion has motivated its use for reactive robot control. Most commonly, time-to-contact has been applied to collision avoidance (*e.g.,* [Nelson and Alloimonos 1989][Coombs et al. 1998]). Few, however, have applied the cue to tasks such as docking (or landing). Visually controlling velocity to achieve close proximity docking with a looming surface (without collision) requires significantly greater tolerance to noisy on-board conditions. This is because of the significantly higher risk of collision as surface distance decreases, but also because of increased susceptibility to external forces such as bumps, undulations and wind effects as forward velocity is reduced. Existing visual docking and landing strategies typically assume specific camera motions and/or surface orientations (or at least assume these to hold) during the execution of the task (*e.g.,* [Cipolla and Blake 1997]). This limits the applicability of such approaches under real-world conditions.

1.1.5 Systems of visual control using visual motion

The direct use of time-to-contact estimation for motion guidance represents a *visuomotor* system of control. Extracted visual quantities are used directly to guide motion, thereby avoiding the need for internal reconstructions of the scene [Lee 1980]. Thus, the choice of visual cues are necessarily task specific, exploiting the visual conditions that characterise the successful execution of the task.

An alternative approach is to derive control schemes directly from explicit estimates of observer self-motion and a reconstructed representation of the scene. This represents the traditional approach to vision-guided navigation in computer vision, referred to as *structure-from-motion*. Structure-from-motion is a key and active area of research in computer vision. A central motivation for this approach is that the obtained solutions may support a wide range of navigation tasks, thus providing a potentially more general solution. A major drawback in applying structure-from-motion to visual control is that this requires solving for all structure and motion parameters in each iteration of the control loop, placing significant computational demands on the system. For many visual control tasks, full structure-from-motion recovery is not required.

Such choices of visual control systems highlight the varying roles prescribed to vision. How vision best serves the needs of perception and motion control remains a fundamental question for anyone seeking to understand, or build, visual perception and navigation systems.

1.1.6 Projective geometries for visual contact estimation

Whether for reactive visuo-motor control or explicit structure-from-motion recovery, the projective geometry in which visual information is expressed has significant implications for the extraction of visual data, and the design of visual control schemes. Ecological studies [Gibson 1979], and subsequent theoretical analysis [Fermüller and Aloimonos 2000], have made clear that the geometry of eye designs in the natural world have a significant impact on the visual navigation strategies employed. In robot vision, however, a pinhole perspective projection model is most commonly applied, a choice primarily dictated by the use of standard digital cameras.

Recent theoretical examination of alternative eye geometries such as the compound eye of flying insects, suggests a spherical projection model (as an approximation to this geometry) provides a better suited alternative [Brodsky et al. 1998; Fermüller and Aloimonos 1998]. In particular, researchers have identified potential advantages for egomotion recovery [Nelson and Aloimonos 1988; Lim and Barnes 2008], and for the unambiguous recovery of structure-from-motion parameters [Brodsky et al. 1998; Fermüller and Aloimonos 1998]. Understanding how this choice impacts on control design, the choice of visual inputs used for control, and the strategies employed to extract them, is essential to the development of robust navigation strategies based on visual motion.

1.2 Core contributions of this thesis

In this thesis we propose new visual cues and algorithms for estimating potential surface contact from optical flow. We explore this primarily in the context of robot navigation and perception. Through this study, this thesis offers the following contributions.

A robust algorithm for docking a mobile robot using optical flow field divergence

We have developed an algorithm to compute the flow field divergence, or time-to-contact, in a manner that is robust to small rotations of the robot during ego-motion. We achieve this by tracking the *focus of expansion* of the optical flow field and using this to compensate for ego rotation of the image. This operates without the need for explicit segmentation of features in the image, using complete gradient-based optical flow estimation in the optical flow computation.

A unified algorithm for landing/docking using optical flow field divergence under spherical projection

We propose a single unified strategy for performing controlled approaches to planar surfaces of arbitrary orientation. Central to this is the use of optical flow field divergence under spherical projection, which allows time-to-contact to be measured for an arbitrary angle of approach, without explicit knowledge of the surface orientation, and without de-rotation of the flow field. The proposed scheme provides the first general solution to the docking/landing problem using time-to-contact.

A strategy for estimating 3D depthmaps from spherical optical flow in real-time

We present a strategy for generating real-time relative depth maps of an environment from optical flow, under general motion. We achieve this using an insect-inspired hemi-spherical fish-eye sensor, and a de-rotated optical flow field. De-rotation is achieved through explicit egomotion estimates obtained using an algorithm first proposed by Nelson and Aloimonos [1988] for use on a full view sphere. For the first time, we demonstrate the application of this algorithm over real image sequences, to support real-time structure-from-motion recovery.

A technique for estimating time and location of impact based on primate vision

We present a preliminary investigation for the use of optical flow to predict the time and location of impact of an incoming object. By examining patterns of optical flow, we make predictions on an object's trajectory with respect to a stationary observer, and its time-to-contact with the observer's (assumed planar) body. This approach is modelled on the observed behaviour of neurons in the F4 region of the pre-motor cortex of primates.

A flow-based hazard alert system for classes of on-road hazards

We report preliminary results from work towards the development of low-level visual motion cues to identify potential hazards during on-road driving. In conjunction with a clinical study of hazard perception in older age drivers, we consider the detection of a range of hazardous scenarios identified as particularly challenging for older drivers. We present results obtained using the same hazard perception test that will be used in clinical trials.

1.3 Thesis Overview

This thesis is structured as follows:

Chapter 2 outlines the motivation and approach upon which the contributions of this thesis are based. We contrast the different roles prescribed to vision in the context of robot navigation. We argue that biology provides an important and informative base for developing visual navigation behaviours. Additionally, we motivate the use of optical flow as a primary sensory input for perception and motion control.

Chapter 3 overviews the estimation of optical flow and examines the underlying theory of inferring scene structure and self-motion from optical flow. We also provide an overview of structure-from-motion techniques for visual navigation, and motivate consideration of a spherical projection model to address inherent limitations of traditional structure-from-motion techniques.

Chapter 4 provides an in depth review of techniques for estimating and applying time-to-contact as an input to visuo-motor control schemes. We particularly focus on its application to visuo-motor landing and docking, where we highlight specific issues and limitations associated with its use for fine motion control tasks.

Chapter 5 describes a visuo-motor control strategy for docking a mobile robot with upright, near fronto-parallel surfaces using time-to-contact estimates. We derive a time-to-contact estimator that is robust to small rotations of the robot inevitably introduced during egomotion. We validate the theoretically derived estimator with quantitative open-loop simulation and real image experiments. We then integrate the time-to-contact estimator into the control loop of a mobile robot performing close proximity docking manoeuvres with an upright surface.

Chapter 6 presents a generalisation of the docking scheme presented in Chapter 5, allowing a mobile robot to dock with (or land on) a surface of arbitrary orientation. Central to this strategy is the use of a spherical projection model, over a wide field of view. We show that distinct advantages are gained if time-to-contact is estimated from flow divergence on the view sphere, and from this, derive a scheme that exploits the global divergence maximum across the projected surface. The viability of the proposed scheme is tested in open-loop experiments over a range of image sequences. Closed-loop simulation experiments, and on-board trials examine the in-system performance of the control scheme.

Chapter 7 examines optical flow under a structure-from-motion framework, using a spherical projection model. We propose a scheme for generating 3D relative depthmaps of the environment, in real-time, from optical flow under spherical projection. We present the details of the scheme, and discuss its application to generating 3D depthmaps on the view sphere. We implement the proposed scheme for use with a wide-angle hemispherical sensor, and quantitatively and qualitatively assess performance in simulation, and over real image sequences.

Chapter 8 considers visual contact estimation for self-moving objects and a stationary observer. We briefly review background literature in primate neuroscience which then motivates a proposed time and location of impact prediction scheme. We report preliminary results and a discussion of future directions for this research.

Chapter 9 presents a heuristic-based contact estimation scheme for the detection of non-looming side-entering on-road hazards. We briefly overview motivations drawn from studies of visual ageing and its effects on driver hazard perception. We then outline a scheme for the in-car detection of other side-entering objects. We present preliminary results and discuss future work for the project.

Chapter 10 sets out the overall conclusions and outcomes of this thesis. We also list the limitations of this study, and discuss future work.

Chapter 2

Motivation and Approach

2.1 Introduction

In this thesis we explore vision as a primary sensor for navigation and perception. Before presenting the novel contributions of this thesis, we provide background and a review of relevant literature. We split this review into three parts. The first part (this chapter) motivates our approach to visual contact estimation, providing a philosophical basis for the work presented in this thesis. The second part (Chapter 3) provides a review of literature and theory on the estimation and traditional use of optical flow to infer scene structure and self motion (*i.e.,* structure-from-motion). The third part (Chapter 4) motivates the use of time-to-contact from optical flow as an alternative to structure-from-motion recovery, comprehensively reviewing previous work in the estimation and application of time-to-contact for collision avoidance, docking and landing.

We now outline the approach to visual contact estimation and navigation adopted in this thesis. In so doing, we review the defined roles prescribed to vision when serving the needs of navigation, and in particular, for the control of motion. We explore this in the context of robot navigation systems, for which the choice of visual cues utilised, the processes by which they are extracted, and the perceptual purposes they serve vary widely. Through this, we argue for the importance of visual control strategies that avoid global maps or scene reconstructions. We further motivate biological vision as a useful and informative base for the design of reactive visual navigation and perceptual algorithms, and review biological arguments in support of visual motion as an important sensory input for perception and motion control.

The chapter is structured as follows. Section 2.2 considers perception as it is defined for robot navigation, splitting the perception problem into two roles: deliberative navigation and reactive control. Section 2.3 outlines the classical methodology of computer vision for navigation and perception where its traditional role has been viewed as a process of 3D reconstruction. In Section 2.4 we explore how vision has been applied in existing vision-guided robot navigation systems, emphasising the distinction between vision for map-based deliberative navigation and for reactive motion control without global maps. Section 2.5 then presents arguments from biological visual perception in favour of visual control without world models, highlighting the central role of visual motion in biological vision and motion control. Section 2.6 then motivates visual motion for robot navigation and control. In addition, we justify the choice of optical flow estimation for recovering dense visual motion across the field of view under real-time constraints. Finally, Section 2.7 summarises the chapter.

2.2 Perception for robot navigation

The design of perceptual architectures for robot navigation has undergone a significant paradigm shift over the last thirty years. In this, the role of perception has moved from a process supporting purely *deliberative navigation* based on symbolic representations of the scene, to a methodology encompassing non-deliberative, *reactive navigation*. This shift has paved the way for significant advances in robot navigation systems. It is therefore important to understand why this paradigm shift has occurred, and the implications of this on the defined role of vision for perception and navigation.

2.2.1 The sense-plan-act paradigm

The dominant architecture of early robot navigation systems was the *sense-plan-act* architecture [Arkin 1998]. This architecture is characterised by a sequential pipe-line of modular processes, taking sensory input to output commands of actuators. The *sense* module receives all sensory inputs, from which a 3D world model is maintained. The *plan* module interprets this model, and in conjunction with the set goals of the system, composes sequences of actions to achieve each goal. The *act* module executes the plan via appropriate outputs to actuators.

The sense-plan-act architecture was well aligned with contemporary artificial intelligence approaches of the time, relying heavily on the use of symbolic representations of the world for reasoning and action planning [Arkin 1998]. Thus, in order to plan and actuate movement, an accurate internal model of the environment must first be obtained.

The earliest examples of such systems include the mobile robot *Shakey* (Stanford Research Institute) [Nilsson 1969; Nilsson 1984], and later work by Giralt *et al.* [1979], and Moravec [1983]. Moravec was the first to demonstrate a mobile robot (*The Stanford Cart*) navigating autonomously using a full geometric model without an *a priori* model.

This pioneering work provided significant new insights into the complexity of autonomous navigation. However, systems developed under the sense-plan-act paradigm generally lack the robustness required for real-world application [Arkin 1998]. Most problematic is the strict requirement for the construction and maintenance of accurate 3D world models before reasoning and actuation may occur. As a result, delays and errors introduced through the scene reconstruction process are propagated through subsequent planning and motor actions. High computational costs in maintaining internal models also place significant restrictions on the speed of robot motion and complexity of working environments. For these reasons, the architecture has been largely abandoned.

2.2.2 Behaviour-based navigation

The behaviour-based paradigm asserts that centralised control and global world models are not necessary to achieve autonomous robot navigation [Arkin 1998]. Moreover, reliance on such models in the control loop may be detrimental to the overall robustness of the system when environments are complex and dynamic. Intelligent behaviour may instead be realised through the complex interplay of low-level modules, and the environment. These *emergent behaviours* form the basis of Brook's *subsumption architecture* [Brooks. 1986; Brooks 1990]. This architecture represents the first significant shift away from the sense-plan-act model.

Fundamental to behaviour-based navigation is the decomposition of the general navigation problem into a hierarchy of navigation tasks (behaviours). At the lowest level reside the most basic reactive tasks (*e.g.,* obstacle avoidance, free-space navigation). At higher levels reside processes to compute goals and coordinate the choice of low-level behaviours.

The subsumption architecture has been applied to many successful robot navigation systems. For example, Horswill [1993] applies the architecture for the navigation of a mobile robot tour guide. Brooks and Stein [1994] use a subsumption architecture as part of a humanoid robot (from the waist up) to test theories of robot-human interaction. Cheng and Zelinksy [1998] describe a behaviour-based navigation system capable of coordinating multiple navigation subsystems. Lenser *et al.* [2002] report on the implementation of a behaviour-based framework for soccer playing robots. More recently, Hentout *et al.* [2007] describe a three level behaviour-based system for navigating a mobile manipulator robot.

2.2.3 Visual perception under a behaviour-based framework

Despite significant advances achieved under the behaviour-based paradigm, vision-guided navigation remains difficult. While behaviour-based navigation systems remove the requirement for global world models as a pre-requisite for action, the methodology does not replace the need for high level deliberate executive function. Thus, visual perception under a behaviour-based architecture must support both the needs of deliberative and reactive navigation. How vision best serves these needs remains a fundamental question for both the computer vision and robotics community.

2.3 Classical computer vision for navigation and perception

In the previous section we defined two dominant roles for perception in robot navigation: perception for deliberative navigation and for reactive control. We now discuss the perspective of classical computer vision and the approach this implies to visual perception for navigation. We also outline the traditional approach to 3D reconstruction in computer vision, and discuss reasons why the classical formulation of the problem is ill-suited to the needs of real-time visual navigation tasks.

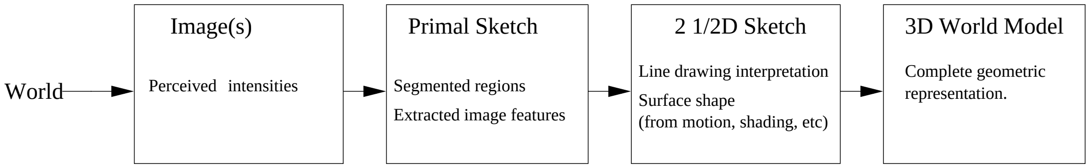

Figure 2.1: Marr's representational framework for vision processing, defined as a pipe-line of sub-processing stages to yield a complete scene reconstruction.

2.3.1 Marr's theory of computational vision

The classical role of vision in navigation and perception is exemplified by Marr's theory of vision [Marr 1982]. Marr argues that vision is best understood as an information-processing problem, emphasising the need for internal representations to support the information extraction process. This is realised through a sequence of intermediate representations, beginning with a 2D array of image intensity values, to produce 3D, object-centred descriptions. Thus, Marr defines vision as a process of scene reconstruction. Figure 2.1 shows the pipeline of vision processes forming Marr's representational framework.

Marr's theory of vision is widely regarded as the first formalisation of a scientific methodology for computer vision [Barnes and Liu 2004]. It also fits naturally with the sense-plan-act perceptual architecture of early robot navigation systems, and in general, serves the needs of deliberative navigation and perception. While aspects of the theory have been subsequently discarded, modern computer vision remains heavily influenced by the work of Marr. In particular, geometric 3D reconstruction remains a dominant focus of computer vision research.

2.3.2 Traditional 3D scene reconstruction in computer vision

The task of inferring the 3D structure of an environment from multiple 2D images of the scene has been a topic of significant interest in computer vision for over thirty years. The problem is classically defined as that of taking point correspondences between two or more views to infer the 3D location associated with each point correspondence [Hartley and Zisserman 2000]. In the simplest case, point correspondences between two overlapping images provide the input. Where the displacement of these points is the result of camera motion, the problem is commonly referred to as the *structure-from-motion* problem.

Given a sufficient number of such correspondences, the complete projective geometry of the camera pairs can be computed, allowing the 3D location of the projected points to be inferred via triangulation (at least up to scale). This is achieved by estimating the *fundamental matrix*, providing linear constraints on each correspondence. A common approach to estimating the fundamental matrix is via the *eight-point algorithm* [Longuet-Higgins 1981].

For general scene reconstruction over multiple views, the problem is significantly more complex. In this case, the dominant methodology in computer vision is *bundle adjustment* [Triggs et al. 2000]. Techniques that employ bundle adjustment attempt to fit a non-linear model over point correspondences, typically over many images. This is performed as an iterative process, and thus requires an initialisation step prior to its execution. Recent examples of the classical reconstruction approach include Pollefeys *et al.* [2004] who produce complete 3D rendered models from hand-held video sequences, and Vidal and Hartley [2008], who propose a three-view reconstruction technique capable of handling multiple motions in the scene.

Techniques for obtaining dense 3D reconstructions from multiple views have advanced significantly over the last twenty years. However, this classical approach is generally regarded as infeasible for real-time closed-loop control of a robot. Most problematic are the computational demands associated with constructing and optimising world models in complex environments. This is an inherent drawback of the methodology. While efficient methods for computing bundle adjustment exist [Triggs et al. 2000] achieving real-time performance currently requires significant reductions in the detail of models, and/or fusion with other sensor measurements for camera positioning (*e.g.,* GPS, INS) (discussed in more detail in Chapter 3). We therefore do not apply classical reconstruction-based structure-from-motion in this thesis.

2.4 Vision-guided robot navigation

While classical computer vision techniques have proven difficult to apply in real-time robot navigation systems, there exist many examples of vision being successfully applied to various robot navigation tasks. In this section we review existing approaches to vision-guided robot navigation. We divide these approaches into three classes:

1. visual navigation with global maps,
2. visual navigation with local egocentric maps, and
3. visual navigation without maps.

Through this spectrum of approaches we emphasise the important distinction between visual perception for deliberative navigation, and visual perception for the direct control of motion.

2.4.1 Visual navigation using global maps

Dense geometric reconstructions are rarely applied in robot navigation systems. Rather, simpler, more efficient scene mapping frameworks that better support the needs of deliberative navigation tasks such as path-planning, localisation and visual odometry are employed. These are typically defined independently of the sensing technology, assuming only that an estimate of range to surfaces is available. An advantage of this is that measurements from multiple sensors may be fused together and used to construct a more accurate model. We briefly overview two popular approaches to map-based navigation below, with specific examples of their use in visual navigation.

2.4.1.1 Grid-based mapping

Grid-based mapping divides the world into discrete cells, with each cell conveying the traversability of the space it represents. One of the earliest and most popular grid-based mapping frameworks is *Occupancy grids*. Occupancy grids provide a probabilistic framework for fusing multiple sensor readings into surface maps of the environment [Thrun et al. 2005]. Moravec and Elfes [1985] proposed a technique for encapsulating uncertainty by associating each cell with a probability of occupation. An occupancy grid at a given time is estimated as a posterior probability over a set of possible maps. This is generated from the set of scene measurements (surface depths), and robot poses with respect to a world coordinate frame, over time. Thus, the map generation assumes the robot's path is known. When applied to robot navigation, occupancy grids are most commonly projected onto the ground plane, representing a 2D slice of the 3D scene structure. Examples of vision-based navigation systems utilising occupancy grids include Murray and Little [Murray and Little 2000], who apply real-time stereo-vision disparity analysis to construct a 2D occupancy grid. More recently, Correa and Okamoto [2005] generate an occupancy grid using omni-directional stereo vision to compute depth in the scene.

Other mapping frameworks such as *artificial potential fields* [Khatib 1986] have also been applied to map-based navigation. In this, range estimates of surfaces are used to form a 2D vector field, where each vector represents the combined forces of attraction towards a goal, and repulsion away from obstacles. Such representations may facilitate path-planning within a global map (*e.g.,* [Warren 1989; Urmson et al. 2002]), or can be applied within egocentric maps for reactive navigation (*e.g.,* [Haddad et al. 1998]).

2.4.1.2 Simultaneous localisation and mapping (SLAM)

Map-based navigation in an unknown environment requires both building a map and localising within that map. This, however, represents a circular dependency in that both tasks imply solutions to the other already exist. This key issue in map-based navigation forms the basis of the *Simultaneous localisation and mapping* (*SLAM*) problem [Smith and Cheesman 1987; Durrant-Whyte 1988]. The use of vision for SLAM (Visual SLAM) represents the closest thing to geometric 3D visual reconstruction in broad use in robot navigation.

Interest in SLAM grew dramatically when Csorba and Durrant-Whyte [1997] proved convergence if scene mapping and localisation are combined into a single estimation problem. Thus, solutions are obtained by measuring the relative position of landmarks with respect to the robot, and correlating subsequent observations of the same landmarks over time (while also adding new landmarks as they appear). Most commonly, solutions are obtained by computing a probability distribution describing the joint posterior density of landmark observations and the robot pose at each discrete time instant [Durrant-Whyte and Bailey 2006]. By recursively incorporating new landmark observations over time, correlations between landmark estimates increase monotonically, thus ensuring estimates of landmark locations can only improve with more observations [Dissanayake et al. 2001].

Two dominant representations exist for computing solutions to the SLAM problem: Kalman (or extended Kalman) filtering (KF or EKF), and particle filtering. KF- and EKF-SLAM employ a state-space model, representing landmarks as a joint set of covariances and assume Gaussian disturbances in the robot motion and landmark observation model [Dissanayake et al. 2001]. Particle filtering models robot movement via samples of a non-Gaussian probability distribution [Thrun et al. 2000; Montemerlo et al. 2003]. Landmarks are typically represented as a set of independent Gaussians rather than joint correlations, providing significant efficiency gains [Durrant-Whyte and Bailey 2006].

A wide range of sensors, including laser range sensing (*e.g.,* [Thrun 1998]) and sonar (*e.g.,* [Leonard et al. 2002]), have been applied to implementations of SLAM solutions. Increasingly, focus has turned to visual SLAM techniques. In many cases, stereo matching is applied to estimate depth in the scene. Recent examples of this include Elinas *et al.* [2006], and Dailey and Parnichkun [2006]. While visual SLAM algorithms are typically designed for indoor or structured outdoor environments, recent work such as Marks *et al.* [2008] have applied visual SLAM in unstructured outdoor environments. Other recent examples of real-time visual SLAM techniques include [Gee et al. 2008], [Davison et al. 2007] and [Milford and Wyeth 2008].

2.4.1.3 Summary of global map-based visual navigation

Global map-based methods are well suited to deliberative navigation tasks such as path planning and map building. However, the high computational demands of constructing and maintaining global maps in complex scenes makes them ill-suited for use in the control loop. Where vision is in direct control of motion, the emphasis of visual perception is typically on the recovery of local egocentric information rather than global mapping.

2.4.2 Visual navigation using local maps

Visual navigation from local maps represents a bridge between mapless reactive control and deliberative navigation from global maps. A local map is typically used to provide an egocentrically defined representation of the environment distinguishing traversable space from obstructed space in the immediate area. Thus, the primary task of vision is most often to compute the range to obstacle surfaces across the field of view (we refer to this as a *depth map*). However, in contrast to global map-based techniques, no attempt is made to consolidate local maps into a world coordinate frame.

Given a depth map, grid-based techniques such as occupancy grids (introduced in Section 2.4.1.1) provide a natural representation for local mapping and path planning. Vision-based examples include Otte *et al.* [2007], who acquire real-time depth maps from stereo disparity to create an egocentrically defined occupancy-grid. The occupancy grid is defined entirely within the image space, allowing path planning to take place within the image. Pacheco *et al.* [2008] apply depth-from-focus using a monocular vision system to generate an egocentrically defined occupancy grid for navigation in indoor environments.

Local depth maps have also been used to form *Vector field histograms*, whereby a one-dimensional discretised polar obstacle density function is defined [Borenstein and Koren 1991]. A robot's heading is determined via a search over an obstacle density function. The direction with lowest obstacle density closest to the target direction is selected. Extensions of this method have since been proposed to incorporate robot size and dynamics (known as VFH+ [Ulrich and Borenstein 1998]), and the incorporation of A* searching to verify a heading choice with respect to the goal (known as VFH* [Ulrich and Borenstein 2000]).

Local mapping from vision-based depth recovery has also been applied to off-road navigation in outdoor environments. Competing in the 2005 DARPA *Grand Challenge*, the autonomous ground vehicle, *TerraMax* [Caraffi et al. 2007] completed the race using a vision-based obstacle detection system. A stereo rig with variable baseline is utilised to provide accurate depth estimates for both distant surfaces and objects in close proximity. By correlating edges between stereo views, a vertical disparity map

(V-disparity) is acquired, from which depth in the scene is inferred. Using models of ground plane curvature, obstacles in front of the vehicle are identified and mapped to a 2D egocentric coordinate system. Stereo-based disparity is also applied for assessing terrain traversability and obstacle avoidance on the current NASA/JPL Mars Exploration Rover mission [Matthies et al. 2007; Olson et al. 2007]. Feature correlation between stereo pairs is also used to acquire egomotion and odometry information.

2.4.2.1 Summary of local map-based navigation

The removal of the requirement for registering local maps in a global coordinate system provides a significant efficiency gain over global-mapping techniques. The recovery of detailed egocentric maps, however, still requires significant computation, thus hindering their use in the control loop. While hardware and algorithm advances make their use in the control loop increasingly plausible, local maps are most often used for local path planning to support lower-level reactive behaviours (*e.g.,* to avoid local minima).

2.4.3 Visual navigation without maps

Visual navigation techniques that do not employ maps span an array of approaches, ranging from those that exploit structures and features in the scene, to those that derive motion control schemes within the image space itself using directly measured visual quantities. In most cases, vision is used exclusively to support reactive motion control schemes. We outline some of these approaches below.

2.4.3.1 Visual navigation in semi-structured environments

The operating environment of many robots contain characteristic structural properties that may be exploited to simplify visual navigation. Conventional indoor environments, for example, are typically dominated by planar surfaces. In this context, Zhou and Li [2006] propose a technique for extracting the ground plane from a single camera. To detect the ground plane, they apply a homography-based approach through the examination of tracked image features. From this, a dominant homography between two frames is extracted. Dao *et al.* [2005] present a similar approach whereby lines describing planar features are tracked and used to compute a homography between frames. By combining odometry data and various heuristics exploiting the camera-robot configuration, the ground plane is segmented allowing on-ground obstacles to be detected. Micusik *et al.* [2008] present a method for detecting orthogonal planar surfaces using a single camera via estimates of the vanishing points in the three orthogonal directions. A probabilistic approach is then adopted to estimate planar patches, using a Markov Random Field (MRF) and a colour-homogeneity heuristic.

In the context of outdoor environments, visual navigation is made more challenging by the comparatively less structured conditions (in addition to other well reported factors such as variable lighting conditions). In the case of vision-based road navigation, however, specific features such as the road plane and/or the road edge, or lane markings on the road are commonly utilised to maintain a vehicle's path. In the original version of the autonomous vehicle project, Navlab (Carnegie-Mellon University Robotic Institute) [Thorpe et al. 1988], road following is achieved via a combination of colour and texture classification. A Hough-based voting scheme is applied to the extracted road edges to guide steering. More recently, Wedel *et al.* [2008] propose a scheme for on-road navigation that alleviates the commonly applied planar road assumption. From stereo camera data, a non-planar ground surface extraction is achieved via a parametric B-spline model. An optimisation algorithm is then employed to define the road-obstacle boundary. Armingol *et al.* [2007] present numerous visual subsystems for on-road navigation and human driver assistance, including systems for lane-keeping, pedestrian detection and vehicle detection.

2.4.3.2 Visual servoing and appearance-based navigation

The *image-based visual servoing* methodology exemplifies navigation without internal representations. Rather, motion is controlled via a task specific image function defined in an image feature parameter space [Hutchinson et al. 1996]. Velocity control outputs are then obtained via the application of an inverse image Jacobian, defining a linear transformation between the parameter space and the velocity space.

The majority of work in visual servoing has concentrated on the control of *eye-in-hand* robot arm manipulators [Hutchinson et al. 1996]. However, interest has also grown in the use of image-based visual servoing techniques for mobile robot navigation. In this context, the most common approach adopted is to minimise a cost function associated with the current view of the environment, and some target image. Usher *et al.* [2003], for example, demonstrate this approach using an omni-directional vision sensor on a car-like vehicle. Through deliberate movements, the vehicle positions itself in a target position by aligning specific features in the image. Gaussier *et al.* [1997] apply online learning to train a neural network that maps visual cues to motor actions. Using learnt associations between landmarks close to the goal location and actions leading towards this location, the robot is able to compare its current view with the learnt view to generate appropriate movements. Zhang and Ostrowski [2002] present an image-based visual servoing approach to motion planning for a mobile robot within the image plane (thus avoiding global maps). Motion planning ensures image features remain within the field of view of the robot.

Nierobisch *et al.* [2006] propose a scheme for image-based visual servoing to traverse an environment using previously acquired images as landmarks. A pan/tilt camera is employed to actively track features over large distances, thereby reducing the number of landmark images to be registered. Chen and Birchfield [2006] apply a similar teach and replay approach, but employ a significantly simpler control strategy. Image features are detected and tracked during a training run, and *milestone* images are recorded at regular short intervals. During path execution, a comparison of features in the current view with features in the relevant milestone image provides the basis for heading adjustments based on the relative location of features. A left or right adjustment is determined via a winner takes all vote over all correlated features. Other recent examples of similar approaches include [Mochizuki et al. 2007] and [Segvic et al. 2007].

2.4.3.3 Reactive vision-guided obstacle detection and avoidance

Where the objective is explorative navigation, systems generally employ reactive collision avoidance strategies, based on estimates of relative (or absolute) depth of potential obstacles in the view field. Many techniques assume a ground plane, and base obstacle detection on regions of the image that 'disrupt' the planar model. Shao *et al.* [1995], for example, use stereo disparity and a neural network to identify the ground plane. Deviations from the ground plane disparity model encoded in the neural net are treated as potential obstacles. Burschka *et al.* [2002] also model the ground plane in stereo disparity space using the known camera calibration parameters, baseline and camera orientations of a stereo rig. After removing the ground plane, an egocentric obstacle map is formed via an inverse projection of segmented residual disparities onto the ground plane.

Reactive obstacle avoidance has also been demonstrated using more qualitative visual cues measured directly from the image. For example, Horswill's tour guide robot, Polly [Horswill 1993] achieves reactive visual navigation using numerous qualitative cues. Assuming all obstacles lie within the ground plane, and motion is constrained to the ground plane, the heuristic that obstacle depth increases with the height of its projected location in the image is applied. Other visual cues such as background texture, edge detection and vanishing-points are utilised for obstacle avoidance and corridor navigation. Lorigo *et al.* [1997] demonstrate a simple collision avoidance strategy using image intensity gradients, RGB colour and HSV (hue, saturation, value) information to detect obstacles. By combining the results of these heuristic-based detection criteria, estimates of object boundaries are also obtained.

Another commonly applied visual input for collision avoidance is optical flow. We discuss this approach to robot navigation in Sections 2.5 and 2.6.

2.4.3.4 Summary of visual navigation without maps

Progress continues to be made in developing efficient reactive visual navigation systems. It is apparent from the literature, however, that applying vision under this framework remains a significant challenge. In particular, the significant variation in approaches adopted, and the visual cues employed, indicate that we are far from converging on a general methodology. This issue brings into context the broader question of how vision best serves the needs of navigation, a question that has received significant attention in ecological studies of animal vision. This has motivated interest in biological vision as a basis for developing navigation strategies for robots.

2.5 Biological vision for robot navigation

2.5.1 Gibson's theory of direct perception

Ecological studies of animal vision systems provide compelling support for visual navigation without global models. In particular, the pioneering work of Gibson [1950, 1979] who, from a broad study of animals in their environment, argued that the world itself provides its own best model. Gibson [1979] proposed a theory of *direct perception*, prescribing a role for vision defined by the needs of specific navigation tasks and motion control. Thus, movement is controlled directly from visual information expressed through invariants present in the constantly changing image. Vision is therefore a necessarily constant task, seeking to extract appropriate visual cues to facilitate the needs of navigation and motion control. Movement is guided through embedded *visuo-motor* control schemes.

Gibson's theory of direct perception was largely disregarded by visual perception researchers of the time. Marr [1982], for example, argued that Gibson had significantly underestimated the underlying computational processes to support it. Marr went on to formalise such processes in his computational theory of vision (discussed in Section 2.3.1). The lack of success in applying vision under Marr's reconstruction-based approach motivates interest in Gibson's ecological observations, and the central tenants of direct perception.

2.5.2 Active perception

Active perception may be regarded as the emergence of Gibsonian visual perception theory in behaviour-based robot navigation, though the methodology's origins can be attributed with numerous precursory contributions. It defines vision as a searching, explorative task, defined by the needs of specific navigation capabilities. Vision is removed from a centralised role, and instead distributed and embedded in navigation subsystems to form *visual behaviours*. Several researchers have provided significant contributions to the development of the active perception framework for robot navigation.

Early work saw the emergence of active vision systems. Aloimonos *et al.* [1987]

argue that traditionally ill-posed problems such as structure-from-motion become well-posed through active observation (*i.e.,* deliberate camera motions to control the geometric parameters of the sensor). Notably, however, vision is still defined in terms of a reconstruction task. Ballard's [1991] *animate vision* removes the task of reconstruction from vision entirely, asserting that vision is more readily understood in the context of the visual tasks being performed. Ballard emphasises the important role of gaze control as a means of simplifying tasks such as range determination, and camera stabilisation, among others. Aloimonos [1993a] argues that for many navigation tasks, only a partial recovery of scene structure and self-motion is required. Thus, vision is not viewed as a centralised and isolated subsystem, but rather as part of a more complex system, interacting with its environment in specific ways. These contributions, among others, form what is more generally referred to as active perception.

2.5.3 Active perception with visual motion

Gibson [1979] argued that the vast majority of visual information for the guidance of action is made available through the perception of motion (visual motion). Gibson, among others, have identified specific invariants present in the visual motion field that may be utilised to infer scene structure and self motion. These include:

- **Depth-from-motion (motion parallax)**

  Helmoltz [1925] first noted that motion perception conveyed information about the relative depth of surfaces in the scene. Gibson [1950], however, was the first to formally examine the properties of visual motion that provide such cues. He noted that as an observer translates, the apparent relative motion of objects in the scene provides a direct cue of their relative depth. Thus, for a scene undergoing rigid translation with respect to the observer, it is possible to infer the global structure of the scene from the image motion describing this perspective change (referred to as *motion parallax*).

- **Direction of heading**

  Gibson [1950, 1966] observed that the visual motion induced by self-motion moves radially away from a single point in the field of view. This *focus of expansion*

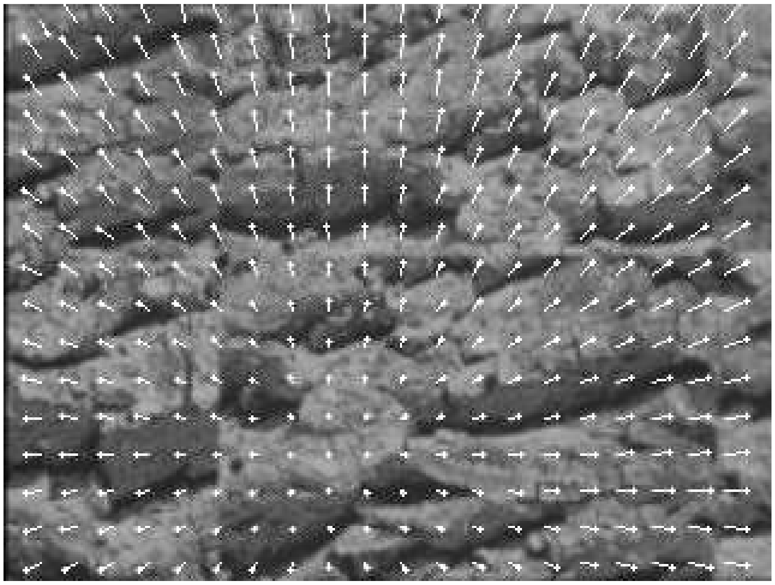

Figure 2.2: The looming effect. Image points diverge from a single point in the image (the focus of expansion).

(FOE) represents a singularity in the flow field (*i.e.,* the optical flow is zero). Gibson argued that the FOE provides a direct cue for estimating heading direction, and thus may be used to regulate and maintain a given heading with respect to the environment. While subsequent psychophysics studies have highlighted issues associated with its general use in perception and navigation (*e.g.,* under high visual rotation during eye movement), it remains an important and useful invariant of the optical flow field. We discuss its role in structure-from-motion and vision-guided robot navigation further in Chapter 3.

- **Visual looming**

  The motion of an observer towards a surface causes an isotropic expansion of the surface image. This image expansion, or *looming effect*, is characterised by motion vectors diverging from the FOE at a rate determined by the ratio of the approach velocity and distance from the surface. An example of this motion pattern is given in Figure 2.2. Visual looming (as measured by the motion field divergence) has been shown to provide a direct estimate of the time-to-contact.

- **Surface orientation/curvature and motion**

Gibson noted the possibility that local orientation, curvature and relative motion of surfaces may be inferred from local examination of the motion parallax [Gibson 1979]. Subsequent theoretical analysis has confirmed this, although the resulting relationships are non-linear (discussed in detail in Section 3.3). Surface slant and curvature may also be inferred from the relative depths of neighbouring points on the same surface.

2.6 Visual motion for robot navigation

For many years, researchers have considered both the estimation of visual motion, and algorithms for extracting structure and motion properties from the visual motion field. In computer vision, the dominant motivation for this has been for solving the structure-from-motion problem (introduced in Section 2.3.2), primarily for 3D reconstruction. However, the direct availability of visual information relating self-motion and scene structure in the visual motion (as observed by Gibson) motivates its use for reactive robot navigation and control. In particular, the efficient use of visual motion by cognitively constrained animals such as insects provides much inspiration for visuo-motor control schemes for robot navigation. We list some example insect-inspired visuo-motor approaches to robot navigation below.

2.6.1 Corridor following

Honeybees achieve centred flight through corridors by balancing the visual motion experienced in opposing sides of their view [Srinivasan et al. 2000]. This has inspired numerous visuo-motor schemes for corridor centring and obstacle avoidance in robot navigation. Coombs and Roberts [1993] use a wide angle, forward-facing active camera to achieve corridor centring. Optical flow is computed in the left and right peripheral thirds of the image. By balancing the maximal flow in both thirds through heading corrections, the robot maintains a centred path between walls and obstacles. Examples of similar corridor following strategies include [Duchon and Warren 1994; Santos-Victor and Sandini 1995; Cole and Barnes 2008].

2.6.2 Visual odometry

Srinivasan *et al.* [2000] showed that the accumulated visual motion experienced by honeybees during flight, provides an accurate estimate for their distance travelled (referred to as visual odometry). For robot navigation, visual odometry offers a viable alternative to traditional odometric gauges which are often inaccurate, or not available such as in flight [Iida 2003]. Weber *et al.* [1996], for example, demonstrate a visual odometry scheme for corridor-like environments. Reduced lateral drift sensitivity is introduced by accumulating the recipricol of flow magnitudes from both sides of the corridor. Iida and Lambrinos [2000] demonstrate visual odometry on an autonomous flying blimp. Flow from the periphery of a down-facing panoramic camera is accumulated over time and used as the odometer.

2.6.3 Altitude regulation, hovering and station keeping

Kelber and Zeil [1997] observed that hovering guard bees maintain a constant distance from the hive entrance through compensatory visuo-motor responses to perceived expansion and contraction of the hive entrance surface. Such observations have inspired strategies for robot hovering and station keeping such as Zwaan *et al.* [2002], who demonstrate a similar approach for an air-based blimp and an underwater robot. Horizontal and vertical positioning is maintained through motor responses directly inferred from the apparent translation and expansion of the surface patch. Roberts *et al.* [2003] report work on a small autonomous helicopter that makes use of stereo image matching and optical flow to measure the height and ground velocity of the helicopter respectively. Other example aerial robot systems that employ visuo-motor control schemes for hovering, either for station-keeping or for landing (in the case of rotor-based robots) include Azinheira *et al.* [2002] and Sharp *et al.* [2001].

2.6.4 Landing and docking

From a study of honeybee landing patterns, Srinivasan *et al.* [2000] propose a model describing the descent velocity profile of honeybees as they perform smooth graze landings. The model proposes that velocity control is achieved by holding the apparent angular velocity of the ground plane constant during the descent, while at the same time reducing the speed of descent proportionally. This visuo-motor strategy yields an exponential decay in forward velocity over time, matching the observed behaviour of honeybees. This model has inspired visuo-motor control schemes for performing controlled approaches towards surfaces. Being a significant focus of the work presented in this thesis, we postpone the discussion of docking and landing techniques until Chapter 4, where we consider visual time-to-contact estimation for achieving such tasks.

2.6.5 Discussion of current visual motion-based navigation

The above work demonstrates how visual motion may be employed to support closed-loop visuo-motor control. However, such control schemes have not been broadly accepted by the robotics community. Of primary concern is the robustness of control schemes based on noisy visual motion estimation. Existing systems primarily serve as a proof of concept, and generally do not provide in-depth consideration of system robustness. For tasks such as corridor centring and visual odometry, control demands are relatively low and thus more tolerant to noisy visual measurements. Where fine motion control in close proximity with surfaces is required (*i.e.,* landing and docking), such issues come to the fore as robustness and efficiency requirements grow significantly. We therefore specifically examine this class of navigation tasks in this thesis.

2.6.6 Measuring visual motion: optical flow versus point matching

Visual motion is most commonly measured in one of two ways, via:

- *point-matching*, the explicit correlation of features in two (or more) images of the same scene, or
- *optical flow*: the estimation of image velocities (*i.e.,* pixels per frame) describing the movement of brightness patterns between two images.

In this thesis we measure visual motion via the optical flow field. We choose optical flow on the basis of its ability to support dense visual motion estimation across large fields of view in real-time. While point-matching, in general, provides a more precise measure of visual motion, optical flow accuracy is comparable over small frame displacements, at a lower computational cost [Lin et al. 2009]. This is due to the linear approximation of motion resulting from its representation as a velocity field rather than as displacements. Our decision to use optical flow is also motivated by its demonstrated role in biological vision, and in particular, insect vision.

2.7 Summary

In this chapter we have reviewed the role of vision for perception in vision-guided robot navigation. We have argued for the importance of visual navigation systems that do not attempt to build and maintain world models in the control loop. Rather, we advocate a role for vision defined by the needs of the specific navigation tasks. Based on ecological observations of natural visual perception systems, we have motivated visual motion as an important and viable input for visual control. Visual motion provides an abundance of visual cues to support such tasks. We have highlighted the two dominant applications of visual motion for navigation: structure-from-motion and visuo-motor control. We have also justified the use of optical flow for dense visual motion recovery over a large field of view, under real-time constraints.

We next review theoretical work in the interpretation of scene structure and self motion properties from the optical flow field. We then consider how such properties have been utilised to serve the needs of visual control under both a structure-from-motion framework, and in Chapter 4, for visuo-motor control using time-to-contact.

Chapter 3

Estimating and Interpreting Optical Flow

3.1 Introduction

We have now explored the role of vision in navigation, and motivated the use of optical flow as a viable sensor for motion control. We considered this in the context of robot navigation, for which the application of vision remains difficult. We now now examine the estimation of optical flow, and the underlying theory of extracting structure and self-motion quantities from the optical flow field. In particular, we focus on visual navigation using traditional structure-from-motion estimation, highlighting inherent issues associated with its estimation and use in the control loop. In the next chapter we consider the use of time-to-contact as an alternative visuo-motor based approach to visual contact estimation for motion control.

This chapter is structured as follows. Section 3.2 derives the optical flow equations employed in this thesis, and reviews literature on the computation of optical flow. Section 3.3 reviews background theory and literature on the inference of structure and motion from local differential invariants of the flow field. Sections 3.4 through 3.6 discuss techniques for estimating egomotion and recovering depthmaps, as well as providing an overview of existing real-time (or close to real-time) structure-from-motion techniques for navigation. In Section 3.7 we discuss issues impeding the accurate and robust recovery of structure-from-motion solutions. Section 3.8 presents arguments in favour of a spherical projection model for structure-from-motion recovery. This is followed by a chapter summary.

3.2 Optical flow

Despite over thirty years of study, the estimation of optical flow remains an active field of research in computer vision. This research has produced a multitude of techniques for recovering image motion, spanning a wide range of approaches and application domains. This thesis does not address the specific issue of optical flow estimation, and thus we do not provide a comprehensive review of optical flow techniques here. Reviews and comparisons of existing optical flow techniques can be found in Beauchemin and Barron [1995], Barron *et al.* [1994], Liu *et al.* [1998], McCane *et al.* [2001], McCarthy and Barnes [2004], and most recently, Baker *et al.* [2007]. Here, we focus on the central issues surrounding the estimation of optical flow for real-time navigation and perception tasks, and provide specific details of the optical flow techniques employed in this thesis.

3.2.1 Derivation of optical flow

This thesis examines optical flow under two projection models: *perspective* (or pinhole projection), and *spherical projection*. We therefore derive the optical flow equations under both models explicitly. Note that the derivations presented below already exist in the literature (*e.g.,* Beauchemin *et al.* [1995] and Fermüller and Aloimonos [1998]). For completeness and consistency of presentation, we re-derive these equations using common definitions employed throughout this thesis. We outline these below.

3.2.1.1 Common definitions

Consider a point, $P = [\begin{array}{ccc} P_x & P_y & P_z \end{array}] \in \mathbb{R}^3$. Let $T = [\begin{array}{ccc} T_x & T_y & T_z \end{array}] \in \mathbb{R}^3$ denote the instantaneous translational velocity of P, and $\Omega = [\begin{array}{ccc} \omega_x & \omega_y & \omega_z \end{array}] \in \mathbb{R}^3$ denote the instantaneous rotational velocity, where Ω is the vector of rotation, and its magnitude, $||\Omega||$, is the angular velocity in radians per unit time. From these definitions, we may define the total instantaneous velocity of P to be:

$$\dot{P} = -T - \Omega \times P, \tag{3.1}$$

The optical flow generated by P is defined as the apparent velocity of P on the image surface. We consider the projection of $\dot{P}$ under both projection models separately below.

3.2.1.2 Optical flow under spherical projection

Let $S \in \mathbb{R}^3$ be a view sphere of radius r, centred on the origin O. The ray originating from O and passing through P gives the ray of projection. The intersection of this line with S marks the spherical projection point, $p \in \mathbb{R}^3$, of P. Thus, the spherical projection of P is given by:

$$p = \frac{rP}{||P||}, \tag{3.2}$$

where $||P||$ is the distance of P from the origin [Ma et al. 2006]. Without loss of generality, we set the radius to one (*i.e.*, $r = 1$).

To obtain the optical flow, $\vec{\mathbf{u}}$, generated by $\dot{P}$ under spherical projection, we take the time derivative of the spherical projection of P such that:

$$\begin{aligned} \vec{\mathbf{u}} &= \frac{d}{dt}\frac{P}{||P||}, \\ &= \frac{\dot{P}||P|| - P\frac{d}{dt}||P||}{||P||^2}. \end{aligned} \tag{3.3}$$

Substituting for $\dot{P}$ we obtain:

$$\vec{\mathbf{u}} = \frac{(-T - \Omega \times P)||P|| - P\frac{d}{dt}||P||}{||P||^2}, \tag{3.4}$$

where p is projection of P on the sphere.

Considering $\frac{d}{dt}||P||$, we note that:

$$\begin{aligned} \frac{d}{dt}||P|| &= \frac{d}{dt}\sqrt{(P \cdot P)}, \\ &= \frac{1}{2}\frac{1}{||P||}\Big(\dot{P} \cdot P + P \cdot \dot{P}\Big), \\ &= \frac{1}{||P||}\Big(-T - \Omega \times P\Big) \cdot P \end{aligned} \tag{3.5}$$

Noting also that $(\Omega \times P) \cdot P = 0$, Equation 3.5 is reduced to:

$$\begin{aligned} \frac{d}{dt}||P|| &= \frac{-T \cdot P}{||P||}, \\ &= -T \cdot p. \end{aligned} \tag{3.6}$$

Substituting Equation 3.6 into Equation 3.4, we obtain the optical flow under spherical projection [Fermüller and Aloimonos 1998]:

$$\begin{aligned} u &= \frac{-T - p(-T \cdot p)}{||P||} - \Omega \times p, \\ &= \frac{1}{||P||}\Big((T \cdot p)p - T\Big) - \Omega \times p. \end{aligned} \tag{3.7}$$

3.2.1.3 Optical flow under perspective projection

Consider again the view sphere S. Let I be a tangent plane (referred to as the image plane) on the surface of S. Without loss of generality, let the tangent plane be centred on the Z axis. Consider again a point $P \in \mathbb{R}^3$, and its radial projection line passing through the origin. Let $p = (p_x, p_y) \in \mathbb{R}^2$ mark the intersection point in the tangent space, I, of the projective line of P through O. The point p is therefore the perspective projection point of P, defined by the well known equations (*e.g.,* [Ma et al. 2006]):

$$\begin{aligned} p_x &= \frac{fP_x}{P_z}, \\ p_y &= \frac{fP_y}{P_z}, \end{aligned} \tag{3.8}$$

where f, the *focal length*, takes the value of r, and P_z gives the depth of the point P assuming the Z axis forms the central axis of projection. We refer to this as the *optical axis*. Note that the image plane resides at $Z = f$, and thus points within the image plane are defined in a 2D coordinate system with origin at the optical axis intersection point.

Let (u, v) denote the two dimensional optical flow field in the image plane, such that:

$$(u, v) = (\dot{p_x}, \dot{p_y}). \tag{3.9}$$

To obtain the image velocities, we take the time derivative of the projection of P in the image such that:

$$\begin{aligned}(u,v) &= \frac{d}{dt}\Big(\frac{fP}{P_z}\Big), \\ &= f\Big(\frac{P_z\dot{P}}{P_z^2} - \frac{P\frac{d}{dt}P_z}{P_z^2}), \\ &= \frac{f}{P_z}\Big(\dot{P} - pT_z\Big). \qquad (3.10)\end{aligned}$$

Substituting $\dot{P}$ for Equation 3.1, and $\frac{P}{P_z}$ for Equations 3.8, we obtain:

$$(u,v) = \frac{f}{P_z}\Big(- T - \Omega \times P - pT_z\Big). \qquad (3.11)$$

Multiplying through by $\frac{1}{P_z}$ we obtain the following equation for the optical flow under perspective projection [Fermüller and Aloimonos 1998]:

$$(u,v) = \frac{f}{P_z}\Big(- T - pT_z\Big) - \Omega \times p. \qquad (3.12)$$

3.2.2 Estimating the optical flow field

3.2.2.1 The aperture problem

The computation of the the optical flow field is difficult. The problem is inherently ill-posed, impeded by the well known *aperture problem* [Hildreth 1984]. Consider a straight-edged surface boundary of constant intensity moving rigidly in an image. If we view a portion of this boundary through a narrow, restricted view, it is impossible to determine a unique solution for the true motion of the edge from the local apparent motion alone. This is demonstrated graphically in Figure 3.1(a). The only motion information we can deduce unambiguously is the component of motion in the direction of the normal to the edge. This is commonly referred to as the *normal flow.* Figure 3.1(b) demonstrates a notable exception. Given two differently oriented edges undergoing the same rigid motion, it is possible to deduce the true motion of the square from the intensity gradients within the aperture.

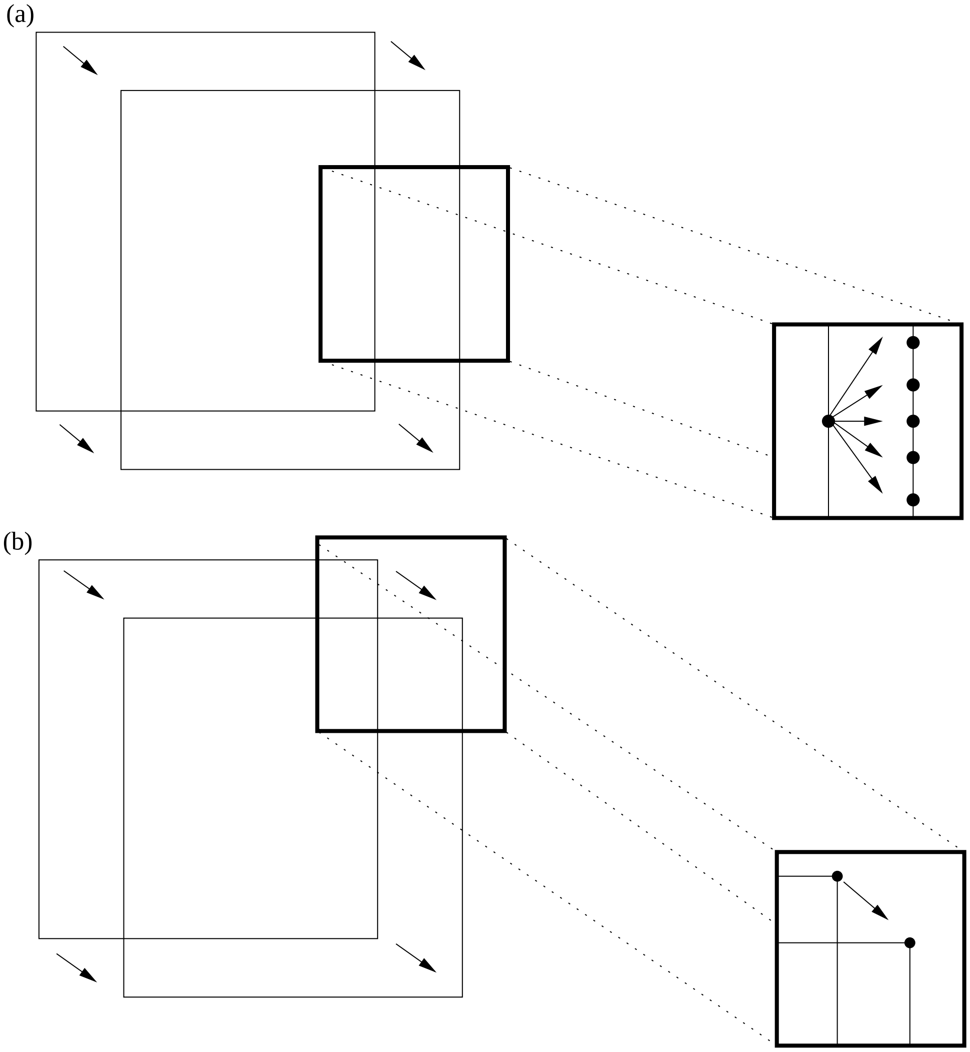

Figure 3.1: The aperture problem: (a) the apparent motion of the straight edge of the square within the aperture can result from any movement of the square with some component in the direction of the normal to the edge, and thus cannot be uniquely determined from this local information alone. (b) the apparent motion of the square corner within the aperture can only result from the true motion of the square, thus providing a unique solution in this case.

3.2.2.2 Classes of optical flow techniques

Numerous optical flow techniques have been proposed, each applying various assumptions and constraints to overcome the aperture problem. Beauchemin and Barron [1995] divide optical flow techniques into four classes:

1. *Energy-based* or *frequency-based methods* estimate optical flow from the output of velocity-tuned filters designed in the Fourier domain. It has been noted that the power spectrum generated from rigid translation of a 2D image pattern lies in a plane in Fourier space [Watson and Ahumada 1983]. Heeger [1988], for example, estimates the optical flow by searching for a plane that best fits the power spectrum of the spatio-temporal signal.

2. *Phase-based methods* estimate image velocity in terms of band-pass filter outputs. Fleet and Jepson [1990], for example, make use of band-pass velocity-tuned filters to decompose the image signal into scale, speed and orientation. More recent examples include Argyriou and Vlachos [2006], and Tho and Goecke [2008].

3. *Correlation-based* or *region-matching methods* search for a best match of small spatial neighborhoods between adjacent frames (discussed further below).

4. *Gradient-based* or *differential methods* use spatio-temporal image intensity derivatives and an assumption of brightness constancy (discussed further below).

Despite their ability to achieve high numerical accuracy, both energy-based, and phase-based methods suffer significant storage and computational overheads in comparison with other approaches [Barron et al. 1994]. On current standard processor speeds, such overheads limit the ability of these techniques to perform under real-time constraints, effectively discounting them from such applications.

Correlation-based methods such as Kories and Zimmerman [1986], Sutton *et al.* [1983], and Little *et al.* [1988] are also computationally intensive, however, attempts have been made to address this issue. Camus [1996, 1997], for example, achieves significant speed-up by relaxing the requirements of sub-pixel accuracy. Rather than searching for matching regions over increasing spatial displacements, Camus proposes searching for matches over time. Thus, the level of temporal support dictates the quantisation level. While the technique is demonstrated on a mobile robot for measuring time-to-contact [Camus 1994] and achieving obstacle avoidance [Camus et al. 1996], subsequent comparisons of its performance for specific navigation tasks have highlighted difficulties in applying it to fine-motion control [McCarthy and Barnes 2004]. The trade-off of execution time for reduced quantisation error is reported to be the most problematic issue for its use in the control loop.

3.2.2.3 Gradient-based optical flow estimation

Gradient-based methods are defined by their use of spatio-temporal intensity derivatives to estimate optical flow. The assumption that image intensity is conserved over time is the basis of all techniques in this category. This is formalised in what is commonly referred to as the *gradient constraint equation*, or *brightness constancy constraint* [Barron et al. 1994]:

$$I_x u + I_y v + I_t = 0, \tag{3.13}$$

where I_x, I_y and I_t represent partial derivatives of the image intensity function $I(x, y, t)$, and u and v represent the horizontal and vertical components of the image velocity at the point (x, y) respectively.

It is immediately apparent that a direct calculation of optical flow is not possible from Equation 3.13 alone. Two components of flow exist for a single point-wise constraint. Thus, the problem is under-constrained, highlighting the inherent ambiguity resulting from the aperture problem. To overcome this, researchers typically exploit the rigidity of surfaces in the scene. This assumption introduces a new constraint on the motion field. Gradient-based techniques apply this constraint in various ways, however, the strategies employed typically fall into one of two categories: *global* techniques and *local* techniques.

Global methods

Global techniques apply a global smoothness constraint to estimate the optical flow field. Horn and Schunck [1981] were the first to demonstrate this approach. They proposed an iterative gradient-based method combining Equation 3.13 with a global smoothness constraint. Optical flow is then estimated by minimising:

$$\int_D (\nabla I(\mathbf{x}, t) \cdot \mathbf{v} + I_t(\mathbf{x}, t)) + \lambda^2(\nabla u^2 + \nabla v^2)\mathrm{d}\mathbf{x}, \tag{3.14}$$

where D is the domain of interest, $\mathbf{v} = (u, v)$, and $\nabla I(\mathbf{x}, t)$ represents the spatial gradients of I about a location $\mathbf{x}$ at time t. λ represents the influence of the smoothness constraint, defined as the sum of the square of the Laplacians of u and v.

Global techniques have two significant drawbacks: (i) the application of global smoothing over object boundaries can cause erroneous flow estimates along surface boundaries; and, (ii) global smoothing implicitly assumes a single motion in the scene (typically due to the camera). While techniques such as Hildreth [1984], Nagel [1990], Alvarez *et al.* [1999], Heitz and Bouthemy [1993], Weickert and Schnörr [2001], and Brox *et al.* [Brox et al. 2004], attempt to alleviate these issues via piece-wise global smoothing (*e.g.,* Nagel [1990] and Weickert and Schnörr [2001]) and/or additional assumptions such as image gradient constancy (*e.g.,* Brox *et al.* [2004]), most are too computationally intensive for real-time application on current hardware.

Local methods

Local gradient-based methods such as Lucas and Kanade [1981], Simoncelli *et al.* [1991] and Weber and Malik [1993] estimate optical flow through the minimisation of constraints over local image regions [Barron et al. 1994]. In the original formulation of the approach, Lucas and Kanade [1981] apply a model of constant velocity as a second constraint on small local neighbourhoods of the image. The model is applied through a weighted, least squares fit of local first-order constraints, such that:

$$\sum_{\mathbf{x} \in \omega} W(\mathbf{x}, t)(\nabla I(\mathbf{x}, t) \cdot \mathbf{v}) + I_t(\mathbf{x}, t)))^2, \tag{3.15}$$

where $W(\mathbf{x}, t)$ denotes a window function and ω is the spatial neighbourhood. From this, a linear system may be defined such that:

$$WA\mathbf{v} = W\mathbf{b}, \tag{3.16}$$

where for k points in the local neighbourhood ω, we define:

$$\begin{aligned} A &= \left[\nabla I(\mathbf{x}_1, y_1), \ldots, \nabla I(\mathbf{x}_k, y_k)\right]^T, \\ W &= \begin{bmatrix} w(\mathbf{x}_1, t_1) & \ldots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \ldots & w(\mathbf{x}_k, t_k) \end{bmatrix}, \\ \mathbf{b} &= -\left[I_t(\mathbf{x_1}, t_1), \ldots, I_t(\mathbf{x_1}, t_1)\right]^T. \end{aligned}$$

Re-arranging Equation 3.16 to solve for $\mathbf{v}$ we obtain the equation:

$$\mathbf{v} = [A^T W A]^{-1} A^T W \mathbf{b}, \tag{3.17}$$

From Equation 3.17 we note that the algorithm requires computing the inverse of $\left[A^T W A\right]$ (where it exists). This, however, is only a 2×2 matrix, defined as:

$$A^T W A = \begin{bmatrix} \sum W^2(\mathbf{x}_k) I_x(\mathbf{x}_k)^2 & \sum W^2(\mathbf{x}_k) I_x(\mathbf{x}_k) I_y(\mathbf{x}_k) \\ \sum W^2(\mathbf{x}_k) I_y(\mathbf{x}_k) I_x(\mathbf{x}_k) & \sum W^2(\mathbf{x}_k) I_y(\mathbf{x}_k)^2 \end{bmatrix}. \tag{3.18}$$

From this, solutions to the local optical flow, $\mathbf{v}$, are obtained.

3.2.2.4 Advantages of local gradient-based flow estimation

Local methods generally achieve better accuracy compared with global techniques [Barron et al. 1994]. While local techniques are susceptible to inaccuracies at motion boundaries due to the breakdown of local flow models, these errors are contained to local regions, and do not influence surrounding flow vector estimates. The removal of global regularisation of the flow field also improves the efficiency of local methods. Moreover, local methods do not enforce the computation of flow across entire surfaces, and thus provide more flexibility in their use.

Notably, Bruhn *et al.* [2005] propose a hybrid technique by combining local velocity constraints with global regularisation. Such techniques offer a potential means in which to obtain spatially consistent flow using local techniques. This, however, increases computational overheads significantly.

Based on these trade-offs, we apply a local gradient-based optical flow method throughout all experimental results reported in this thesis. We choose Lucas and Kanade's technique on the basis of observations outlined above, and its strong performance in previous optical flow comparisons such as Barron *et al.* [1994], McCane *et al.* [2001], and McCarthy and Barnes [2004]. Of particular relevance to this thesis, McCarthy and Barnes [2004] report strong results from the techniques application to robot navigation tasks such as corridor-centring, and visual odometry.

3.2.2.5 Enhancements and extensions to Lucas and Kanade flow estimation

Numerous variations and enhancements have been proposed for use with Lucas and Kanade's method. In this thesis, the following extensions to the classical Lucas and Kanade technique have been applied.

- **Eigenvalue thresholding**: In Equation 3.17, we may regard $\left[A^T W A\right]^{-1}$ as a covariance matrix for the estimated image velocity, $\mathbf{v}$. Thus, a confidence measure may be obtained for the likelihood of an accurate local flow estimate by examining the eigenvalues of this matrix [Simoncelli et al. 1991]. Barron *et al.* [1994] propose thresholding the magnitude of the smallest eigenvalue of $A^T W A$ for determining where flow is estimated. This, however, is a tradeoff between flow field accuracy and flow field density, and thus is not applied when full optical flow fields are required.

- **Pyramidal flow estimation:** Bouguet [2000] describes the implementation of Lucas and Kanade's technique over a pyramidal representation of input images. For each level of the pyramid, a minimisation is iteratively solved to produce a local estimation of the flow. This is then used to initialise the same process at the next pyramid level. While the algorithm is intended for feature tracking (and thus for a restricted number of image points), it may also be applied to obtain optical flow fields across the image. The technique also performs well in the most recently published comparison of state of the art optical flow methods [Baker et al. 2007]. An efficient implementation of the algorithm is available in the open source computer vision developers library: OpenCV [1]

3.2.3 Summary of optical flow estimation

Optical flow is difficult to estimate accurately. This is, in part, due to noise levels in the underlying image signal, but also because of the assumption of brightness constancy and the inherently ill-posed nature of the problem. Optical flow techniques therefore apply different constraints and assumptions in order to estimate image velocity. In many cases, achieving high accuracy comes at the cost of significant computational overheads, and thus a trade-off between accuracy and efficiency is typically required when applying optical flow to real-time tasks. It is therefore incumbent upon algorithms that seek to apply optical flow, to design suitable mechanisms for handling noisy motion estimation.

3.3 Inferring local structure and motion from optical flow

Significant attention has been given to the task of inferring scene structure and self-motion from the apparent rigid motion of a scene when a camera moves. The problem, as formulated by Longuet-Higgins and Prazdny [1980], is that of estimating the camera's six motion parameters, and the structure of surfaces in the environment. This may then be used to reconstruct 3D models of the environment, or to facilitate autonomous navigation within an environment (with or without reconstruction). In this section we focus on the early theoretical development of the structure-from-motion problem, upon which much of the work presented in this thesis is based.

[1] `http://opencv.willowgarage.com`

3.3.1 Differential versus discrete motion

Structure-from-motion algorithms typically assume one of two motion geometries: differential or discrete. *Differential motion techniques* consider the movement of points in the image over an infinitesimal time period. Thus, motion between frames is considered in terms of a *velocity field*. *Discrete motion techniques*, on the other hand, represents this motion as a displacement field. Most commonly, displacements are obtained via feature matching techniques such as the *Scale Invariant Feature Transform* (SIFT) [Lowe 1999], which provide better accuracy over large baselines. Over small displacements, the distinction between differential and discrete motion becomes negligible [Adiv 1985; Lin et al. 2009]. Given differential motion estimates provide a sufficient accuracy at a reduced computational cost, we consider the structure-from-motion problem in the context of differential motion estimation (*i.e.,* from optical flow).

3.3.2 Differential invariants of visual motion

Koenderink and Van Doorn [1975, 1976] were the first to examine the local properties of the motion parallax field for a piece-wise planar surface in motion. By decomposing local optical flow field patches into *elementary fields*, they show that the local spatial change of the optical flow field can be expressed by the linear combination of these elementary fields. The importance of this decomposition is that locally, optical flow can be characterised in a coordinate-free manner, via *differential invariants*. Before introducing these invariants, we first consider the optical flow equations in the context of a single rigidly moving surface.

3.3.2.1 Optical flow across a rigid surface

Locally, we assume optical flow to result from a single continuous rigid surface in relative motion. For a camera centred coordinate system, let $C \in \mathbb{R}^3$ be a (possibly curved) surface projecting to a local patch in the image plane. We may describe the depth of points on the surface of C by the depth function:

$$Z(X, Y) = Z_o + aX + bY + O_2(X, Y), \tag{3.19}$$

where (X, Y) are points on the surface of C, (a, b) is the depth gradient of C at its intersection with the optical axis, Z_o is the distance to C along the optical axis, and $O_2(X, Y)$ represents second-order derivatives of C.

Longuet-Higgins and Prazdny [1980] introduce the perspective projection equations into Equation 3.19, thereby defining surface depth as a function of the image coordinates:

$$Z(x,y) = \frac{Z_0}{1 - a\frac{x}{f_{\mathrm{x}}} - b\frac{y}{f_{\mathrm{y}}} - O_2(x,y)}, \tag{3.20}$$

where (x, y) are the image coordinates, and f_{x} and f_{y} are focal lengths expressed in pixels. Given a known aspect ratio, we may set these both to 1 without loss of generality.

From this we may express the optical flow as [Longuet-Higgins and Prazdny 1980]:

$$\begin{aligned} u(x,y) &= \frac{(-T_x + xT_z)}{Z_o}\Big[1 - ax - by - O_2(x,y)\Big] + \omega_x xy - \omega_y(1 - x^2) + \omega_z y, \\ v(x,y) &= \frac{(-T_y + yT_z)}{Z_o}\Big[1 - ax - by - O_2(x,y)\Big] + \omega_x(1 - y^2) + \omega_y xy + \omega_z x, \end{aligned} \tag{3.21}$$

where $u(x, y)$ and $v(x, y)$ are the horizontal and vertical components of the flow field.

3.3.2.2 Local flow field structure and decomposition

For the inference of properties from the optical flow field, it is useful to represent the optical flow equation in terms of its partial derivatives. Taking a Taylor expansion about the image origin, and assuming a locally smooth surface, the flow field is commonly expressed in terms of the first order derivatives of the flow field only. This approximation is given by the affine transformation [Subbarao 1990]:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} u_o \\ v_o \end{bmatrix} + \begin{bmatrix} u_x & u_y \\ v_x & v_y \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix}, \tag{3.22}$$

where

$$\begin{array}{ll} u_o = -\frac{T_x}{Z_o} - \omega_y, & v_o = -\frac{T_y}{Z_o} + \omega_x, \\ u_x = \frac{T_z}{Z_o} + a\frac{T_x}{Z_o}, & u_y = \omega_z + b\frac{T_x}{Z_o}, \\ v_x = -\omega_z + a\frac{T_y}{Z_o}, & v_y = \frac{T_z}{Z_o} + b\frac{T_y}{Z_o}. \end{array}$$

The 2×2 matrix in Equation 3.22 defines the *velocity gradient tensor* [Subbarao 1990]. It is in the decomposition of this matrix that the local differential invariants may be derived. Koenderink and Van Doorn [1975] apply the Cauchy-Stokes decomposition theorem [Aris 1962] which stipulates that any 2×2 matrix can be decomposed into the sum of an antisymmetric matrix and a symmetric matrix such that:

$$\begin{bmatrix} u_x & u_y \\ v_x & v_y \end{bmatrix} = \frac{1}{2}\begin{bmatrix} 0 & u_y - v_x \\ -u_y + v_x & 0 \end{bmatrix} + \frac{1}{2}\begin{bmatrix} 2u_x & u_y + v_x \\ u_y + v_x & 2v_y \end{bmatrix}. \tag{3.23}$$

The symmetric matrix can be further decomposed into the sum of the multiples of the identity matrix, I, and a symmetric matrix with zero trace, such that:

$$\begin{bmatrix} u_x & u_y \\ v_x & v_y \end{bmatrix} = \frac{1}{2}\begin{bmatrix} 0 & u_y - v_x \\ -u_y + v_x & 0 \end{bmatrix} + \frac{1}{2}\begin{bmatrix} u_x + v_y & 0 \\ 0 & u_x + v_y \end{bmatrix} + \frac{1}{2}\begin{bmatrix} u_x - v_y & u_y + v_x \\ u_y + v_x & -u_x + v_y \end{bmatrix}. \tag{3.24}$$

Factorising each of the matrices then yields:

$$\begin{bmatrix} u_x & u_y \\ v_x & v_y \end{bmatrix} = \frac{\texttt{curl}}{2}\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} + \frac{\texttt{div}}{2}\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + \frac{\texttt{def}}{2}S, \tag{3.25}$$

where

$$\texttt{div} = u_x + v_y, \tag{3.26}$$

$$\texttt{curl} = -u_y + v_x, \tag{3.27}$$

$$\texttt{def} = \sqrt{(u_y + v_x)^2 + (u_x - v_y)^2}, \tag{3.28}$$

and S is a symmetric matrix of zero trace and determinant -1. Notably, S has eigenvalues of 1 and -1, and mutually perpendicular eigenvectors [Cipolla and Blake 1997].

Following Cipolla and Blake's derivation, a rotation matrix, Q, may be applied such that:

$$S = Q^{-1}\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} Q \tag{3.29}$$

where

$$Q = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}, \tag{3.30}$$

and θ is the angle of rotation.

Substituting back into Equation 3.25, we obtain [Cipolla and Blake 1997]:

$$\begin{bmatrix} u_x & u_y \\ v_x & v_y \end{bmatrix} = \frac{\texttt{div}}{2}\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + \frac{\texttt{curl}}{2}\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} + \frac{\texttt{def}}{2}\begin{bmatrix} \cos 2\theta & \sin 2\theta \\ \sin 2\theta & -\cos 2\theta \end{bmatrix}. \tag{3.31}$$

From this decomposition, the differential invariants: `div` (divergence), `curl` (vorticity), and `def` (the magnitude of deformation, or the shear magnitude [Subbarao 1990]) are obtained. These invariants, all defined in terms of partial derivatives of the affine flow field, are independent of the coordinate system [Koenderink and van Doorn 1976].

3.3.2.3 Relating differential invariants to 3D motion and structure

Each differential invariant is directly related to the 3D rigid motion and structure of the scene. From Equations 3.22, we can express the differential invariants in terms of the 3D motion and tangent plane orientation of a surface such that [Cipolla and Blake 1997]:

$$\texttt{div} = \frac{2T_z}{Z_o} + \frac{aT_x + bT_y}{Z_o}, \tag{3.32}$$

$$\texttt{curl} = -2\omega_z + \frac{(-bT_x + aT_y)}{Z_o}, \tag{3.33}$$

$$\texttt{def}\cos 2\theta = \frac{(aT_x - bT_y)}{Z_o}, \tag{3.34}$$

$$\texttt{def}\sin 2\theta = \frac{(bT_x + aT_y)}{Z_o}. \tag{3.35}$$

Koenderink and Van Doorn [1975] derive the above relationships similarly for the unit sphere:

$$\begin{aligned}
\texttt{div}(\hat{p}) &= -2\frac{T_{\text{perp}}}{R(\hat{p})} - \left(\frac{\nabla R(\hat{p})}{R(\hat{p})} \cdot \frac{T_{\text{par}}}{R(\hat{p})}\right), && (3.36)\\
\texttt{curl}(\hat{p}) &= 2\omega_{\text{perp}} + \left|\frac{\nabla R(\hat{p})}{R(\hat{p})} \times \frac{T_{\text{par}}}{R(\underline{\mathrm{p}})}\right|, && (3.37)\\
\texttt{def}(\hat{p}) &= \left|\frac{\nabla R(\hat{p})}{R(\hat{p})}\right|\left|\frac{T_{\text{par}}}{R(\hat{p})}\right|, && (3.38)
\end{aligned}$$

where $\hat{p}$ is a unit vector in the direction of the point $P \in \mathbb{R}^3$, T_{perp} is the velocity in the direction $\hat{p}$ (*i.e.,* perpendicular to the tangent plane at $\hat{p}$), T_{par} is the velocity parallel to the local tangent plane at $\hat{p}$, and ∇R denotes the depth gradient of the surface about P.

3.3.2.4 Interpreting differential invariants

It is important to understand the geometric meaning of each differential invariant derived above. Below we summarise how each invariant relates to motion and structure in the 3D environment.

divergence (`div`) represents the isotropic expansion (or contraction) of a brightness pattern within a local image patch about the image origin. Any velocity in the direction $\hat{p}$ results in a local expansion of optical flow about $\hat{p}$. The rate of this expansion is proportional to the scaled velocity, and thus reflects the rate of approach of the surface projecting to $\hat{p}$. Figure 3.2(a) shows an example of a pure divergent flow field (*i.e.,* $\texttt{curl} = 0, \texttt{def} = 0$).

curl represents a rigid rotation about the optical axis of the brightness pattern in the neighbourhood of the optical axis. Thus, pure rotation of the scene about the direction of view, $\hat{p}$, produces `curl`. Figure 3.2(b) shows an example of the flow field resulting from pure `curl` (*i.e.,* $\texttt{div} = 0, \texttt{def} = 0$).

deformation (`def`) gives the magnitude of the pure shear of the brightness pattern about $\hat{p}$. This translates to a contraction in one direction, and an expansion in an orthogonal direction. The magnitude of this deformation is determined by

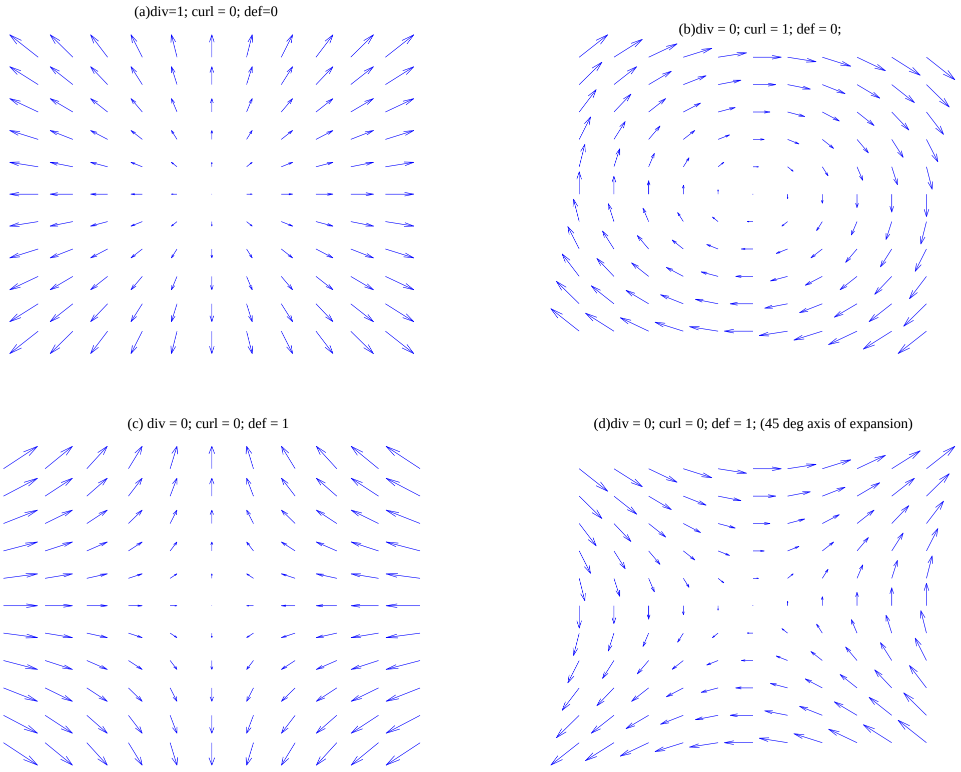
(a)div=1; curl = 0; def=0
(b)div = 0; curl = 1; def = 0;
(c) div = 0; curl = 0; def = 1
(d)div = 0; curl = 0; def = 1; (45 deg axis of expansion)

Figure 3.2: The elementary fields of optical flow: (a) divergence, (b) curl, (c) deformation ($\theta = 0$), (d) deformation ($\theta = 45^o$).

the depth gradient of the surface patch projecting to the neighbourhood about $\hat{p}$ and the scaled velocity of the camera orthogonal to $\hat{p}$ (*i.e.,* parallel to the image plane). The rotation angle, θ, used to form the deformation matrix in Equation 3.31 corresponds with the direction of maximum expansion in the image. Unlike the deformation magnitude, however, this angle depends on the choice of coordinate system. Thus, only the deformation magnitude can be regarded as differentially invariant [Koenderink and van Doorn 1976]. Figures 3.2(c) and (d) show examples of flow fields resulting from pure deformation.

3.3.2.5 Coupling of deformation with divergence and curl

The measurement of `div` and `curl` in the image is subject to additional contributions from the deformation of the local flow field. If fronto-parallel alignment exists, or translational motion is only along the direction of view (*i.e.,* $\hat{p}$), then the deformation term vanishes. However, due to the coupling of these two conditions in the deformation magnitude, it is impossible to account for the contribution of deformation without knowledge of, or assumptions on, the camera motion or local surface orientation. This has significant implications for the extraction of unique quantities relating the local motion parallax to surface structure and motion.

3.3.3 Closed-form solutions to local structure and motion

Early work in structure-from-motion was concerned primarily with developing closed-form solutions to structure and motion parameters from local optical flow samples. In the original formulation of the problem, Longuet-Higgins and Prazdny [1980] consider the use of constraints provided by the partial derivatives of a local flow field patch to solve for the six motion parameters: $T_x, T_y, T_z, \omega_x, \omega_y, \omega_z$, and the local surface gradient (a, b). Given only six affine constraints (Equation 3.22), additional constraints, and/or assumptions are required.

One option is to include second-order constraints from the flow field. Longuet-Higgins and Prazdny [1980], for example, include two additional second-order constraints to provide eight equations. To further constrain the system, they exploit a geometric property that image points lying on the line connecting the image origin and the direction of translation in the image will remain straight in subsequent time-steps. For a non-planar surface, this line will be unique, and thus provides an additional constraint on the translational components of motion.

Waxman and Ullman [1985] employ a system of twelve non-linear equations, including second-order spatial gradients, to resolve the local structure of a curved surface. To avoid solving the non-linear system, a rotation of the image coordinate system is applied. This aligns the image axes with the direction of constant depth on the surface. In addition to removing one of the surface coefficient unknowns, this transformation linearises the constraint equations, allowing more easily obtainable solutions.

3.3.4 The planar surface ambiguity

Reliance on second-order constraints assumes sufficient local curvature of the surface. If the surface is planar, second order gradients vanish, leaving only affine constraints, thus yielding an under-constrained system. Even where curvature does exist, it is unlikely that local curvature would be sufficient to support accurate estimates of second-order gradients [Cipolla and Blake 1997]. Given real-world surfaces are often well approximated by piece-wise planar surfaces, it is important to consider structure-from-motion for the planar case.

Multiple solutions exist if the surface is planar. Inspection of the deformation component of the local flow field decomposition given in Equation 3.38 (and equivalently Equations 3.34 and 3.35 for the perspective case) highlights the existence of dual solutions. Specifically, the contributions of the scaled surface depth gradient, $\frac{\nabla R(\hat{p})}{R(\hat{p})}$, and the parallel translation component, $\frac{T_{\text{par}}}{R(\hat{p})}$, are confounded in the total deformation component of the local flow field. A dual solution is obtained if their direction is interchanged. Figure 3.3 illustrates this duality of solutions. Note that the same duality is expressed in Equations 3.34 and 3.35 via θ, the angle of greatest extension of deformation. This angle marks the bisection of the projected directions $\frac{\nabla R(\hat{p})}{R(\hat{p})}$, and $\frac{T_{\text{par}}}{R(\hat{p})}$ in the image.

Tsai and Huang [1981] were among the first to formally show that multiple solutions to local structure and motion exist when the surface is planar. Subsequent work by Longuet-Higgins [1984], and Waxman and Ullman [1985], have considered both the

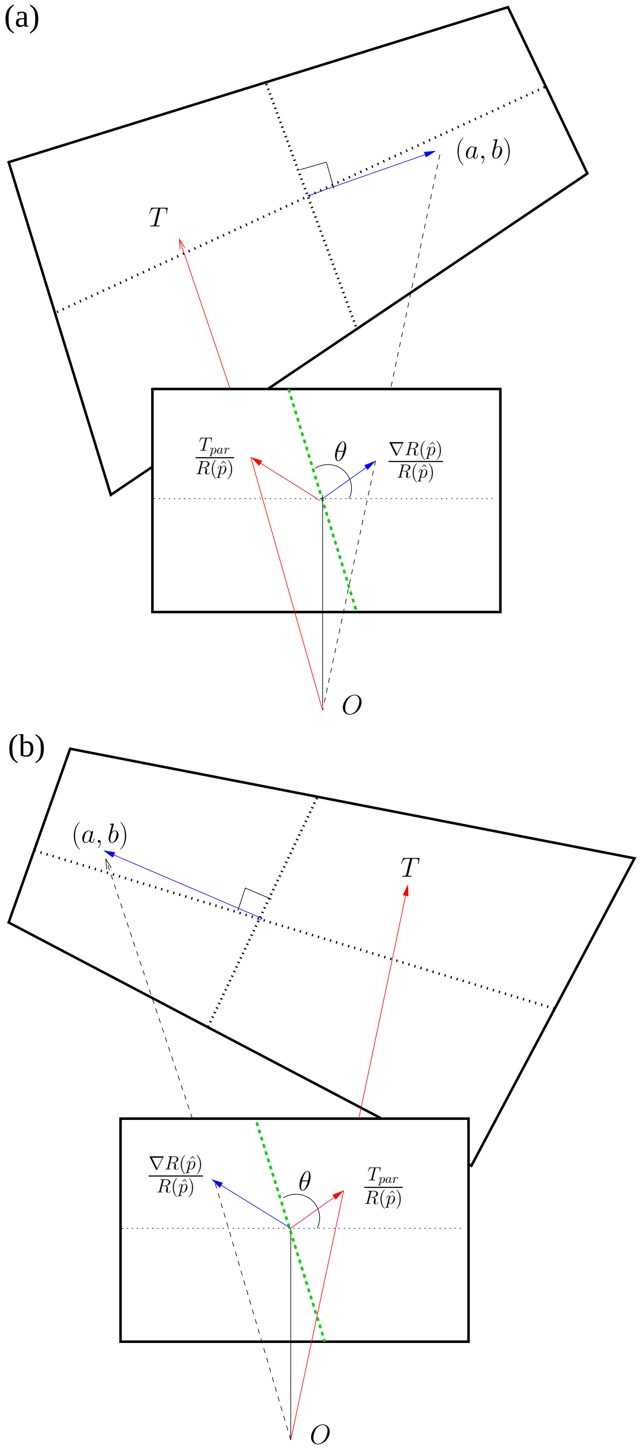
(a)
(a, b)
T
$\frac{T_{par}}{R(\hat{p})}$
θ
$\frac{\nabla R(\hat{p})}{R(\hat{p})}$
O
(b)
(a, b)
T
$\frac{\nabla R(\hat{p})}{R(\hat{p})}$
θ
$\frac{T_{par}}{R(\hat{p})}$
O

Figure 3.3: The planar ambiguity. The same deformation is generated by swapping the direction of motion parallel to the image plane (*i.e.*, $\frac{T_par}{R(\hat{p})}$) with the direction of the surface gradient, as defined by the surface normal (a, b). Note the direction of maximum expansion, θ, is the bisection of the angle between these directions, as shown by the dotted green line.

number of solutions, and the specific cases under which a unique solution can be found, or at least, chosen from a set of possible solutions.

3.3.4.1 Unique solutions for the planar case

In the case that $T_z = 0$, and both T_x and T_y are non-zero, a unique solution is obtained [Waxman and Ullman 1985]. In addition, Equations 3.32 through 3.35 give rise to two other distinct cases. Specifically, unique solutions are obtained when:

1. motion is only along the optical axis such that $T_x = T_y = 0$ and $T_z \neq 0$, or,

2. the surface is fronto-parallel with the image plane, such that (*i.e.,* $a = b = 0$).

In both cases, the components of deformation vanish, thus resolving the surface gradient-translation direction ambiguity.

Notably, a degenerate case exists when $T_x = T_y = T_z = 0$. In this case, the flow field is pure rotation [Waxman and Ullman 1985]. While the motion parameters can be uniquely determined, nothing can be inferred about surface structure.

3.3.4.2 Planar structure-from-motion under general motion

Where conditions for a unique solution cannot be met, researchers have considered alternatives for obtaining workable solutions to local planar structure and motion.

Kanatani [1987] argues that spurious solutions to planar structure-from-motion are removed if a pseudo-orthographic approximation of the optical flow is considered. Under this approximation, the squared focal length terms in Equations 3.21 are omitted. The removal of these terms effectively avoids the spurious solution, which Kanatani shows to be present in the foreshortening effects of perspective projection. If the surface is sufficiently far away, and projective distortion is small, the pseudo-orthographic solution will correspond to the correct perspective solution.

Subbarao [1989] proposes a non-linear spatio-temporal system of equations by combining the affine constraints (Equation 3.22) with temporal derivatives of the flow field. The addition of temporal constraints relating the depth and temporal gradient of the surface patch to the scene motion provide a sufficiently constrained system. The solution assumes smoothness in scene depth, scene motion and deformation, as well as in local patches of optical flow. Under these assumptions, the system can be solved in general, however, a number of degenerate cases exist. Specifically, if the surface is fronto-parallel, or if translation is only along the optical axis or only parallel to the image plane. Subbarao and Waxman [1986] show that unique solutions to the motion and surface orientation parameters can be obtained for a planar surface if either successive optical flow fields depicting the apparent motion of the plane are available, or, two distinct planar patches are available.

3.3.5 Limitations of closed-form structure-from-motion

Closed-form solutions to structure-from-motion are primarily motivated by a need to understand the theory underlying the estimation of scene structure and motion from local properties of the optical flow field. However, there has been little success in applying such techniques to real-time, real-world applications. This can be attributed to a number of factors:

1. the system of equations to solve is typically non-linear, and therefore non-trivial to solve [Aloimonos 1993b],
2. the motion and structure parameters are obtained locally, and generally assume optical flow to provide a close approximation to the true motion field [Verri and Poggio 1989], and
3. the non-linear system is inseparable, forcing motion and structure to be solved simultaneously [Adiv 1985].

Such limitations can be alleviated if structure and motion parameters are estimated from samples of the global flow field (or point correspondences). Specifically, the use of global flow fields increases the availability of optical flow vectors from which to sample, providing greater robustness to measurement noise [Bruss and Horn 1983]. More significantly, egomotion and scene structure can be solved separately [Adiv 1985], providing greater flexibility in the formulation of solutions. For structure-from-motion based navigation, egomotion is typically estimated first, from which scene structure may be directly inferred. In the following sections we discuss techniques for estimating egomotion and scene structure from the global optical flow field.

3.4 Egomotion estimation from optical flow

Numerous optical flow-based approaches have been proposed for estimating egomotion. Techniques typically differ in both the order in which parameters are solved, and in what (and how) features of the optical flow field are combined to obtain the solution.

3.4.1 Egomotion from global flow field samples

Bruss and Horn [1983] propose a least-squares framework for estimating egomotion parameters from a system of seven equations. This is a global scheme which is applied over the entire flow field. While the equations relating camera translation are non-linear, rotation constraints are linear and uniquely expressible in terms of camera translation. A numerical solution (*e.g.,* gradient-descent) is proposed to solve for the translation parameters first.

Adiv [1985] proposes an alternative least-squares framework by subdividing the flow field into patches, and estimating motion parameters for each patch. Optimal local estimates are then combined to form surfaces undergoing the same motion. Adiv applies the same least-squares residual function as Bruss and Horn, but avoids the use of gradient-descent to solve for translation. Instead, a search over the entire solution space of translation directions is applied. Image patches sharing the same solution are then grouped together to form surfaces undergoing the same 3D motion. The technique also handles multiple independently moving objects in the scene.

Heeger and Jepson [1992] form three sets of equations to solve for translation, rotation and scene depth separately. Similar to Adiv, Heeger and Jepson subdivide the image into patches to solve for scene translation locally, and then globally. Two schemes are described for solving rotation. Taking the estimated translational parameters and several samples of the optical flow field, rotation parameters may be estimated from a least-squares minimisation over a linear constraint relating rotation parameters and the flow field. Alternatively, rotation can be solved directly from a large set of linear equations relating 3D motion and depth to the optical flow. This has the advantage of decoupling the estimation of translation and rotation, allowing for parallel computation.

Computational efficiency is the biggest drawback of the techniques described above. In most cases, a least-squares minimisation is required, followed by the application of an iterative numerical technique to solve the non-linear system. An additional issue is avoiding local minima during the minimisation process. In the case of differential motion between two frames, researchers have noted that this is particularly important when camera translation is predominantly forward with respect to the image plane [Chiuso et al. 2000; Oliensis 2005]. In this case, least-squares error functions often contain numerous local minima. This has lead researchers to consider other techniques for robustly estimating egomotion, capable of real-time performance.

3.4.2 Egomotion from the focus of expansion

An alternative approach is to make use of global invariants of the optical flow field. One particular feature of the flow field of use to egomotion estimation is the focus of expansion (FOE). Some of the earliest work in extracting properties of image motion focussed on finding the FOE of the translational flow field.

3.4.2.1 FOE estimation

Given pure translational motion, the FOE is located at the intersection of the resulting radial expansion of flow. Under general motion, however, the location of the FOE cannot be assumed to be the direction of heading, and thus more sophisticated techniques are required to extract the translational FOE for heading estimation.

Jain [1983] proposed one of the first methods for extracting the FOE from matched features in consecutive frames. For each image location, the Euclidean distance to feature locations are separately computed in both frames. The location yielding the largest difference between these sums is taken to be the FOE. This technique, however, assumes noiseless feature correlations.

Considering the extraction of the FOE from the optical flow field, Rieger and Lawton [1985] propose a scheme for estimating heading direction by examining flow vectors at points of significant depth variation. At these locations, differences of flow vectors are computed within local patches, from which an average orientation of difference vectors is obtained. Given sufficient depth variation within the patch, the resulting difference vector orientation will be dominated by the translational component, and thus will be approximately aligned with the translational flow field. A linear least-squares minimisation is applied to estimate the intersection of translational motion vectors.

Li [1992] proposes the use of higher-order flow field derivatives to obtain linear constraints on the translational motion of the camera. A least-squares solution for the translational motion is then obtained from the globally defined constraints. Li notes that by considering the translational motion in the image domain, similar constraints can be obtained to locate the FOE.

Sazbon *et al.* [2004] present a two stage technique for estimating the FOE. A matched filter, F, of specified size (typically 7×7) is applied across the flow image. The FOE is chosen as the image location that minimises a sum of squared difference error function with F. To account for erroneous flow estimates, a weighting function based on the flow vector magnitude is used as a measure of confidence, which may then be thresholded.

3.4.2.2 Structure and egomotion from the FOE

Numerous systems (including some described above) have applied FOE estimation to full egomotion and/or scene structure recovery. In some cases, structure-from-motion recovery is combined with the estimation of the translational FOE providing further constraints on its possible location. Negahdaripour [1996], for example, estimate the FOE by combining spatio-temporal derivatives of the image function with a *depth positivity constraint* applied over candidate depth maps from a set of possible translational motion parameters. The technique exploits a previous result reported by Negahdaripour and Horn [1989] showing that several arbitrarily chosen camera motions provide a strong constraint on the true motion of the camera, and hence the true FOE. Minimisation of the depth positivity constraint enforces the condition that the depth of any scene point projecting into the image must necessarily be positive to be visible. Other techniques such as those described by Joarder and Raviv [1994], and McQuirk *et* *al.* [1998], have also combined depth map estimation and the use of a depth positivity constraint to estimate the FOE. In general, such techniques require sufficient texture across the view field to apply the constraint.

Srinivasan [1999, 2000] estimates the translational FOE in conjunction with camera rotation and inverse depth, but without explicit use of the depth positivity constraint. From a set of candidate FOE locations, a linear system of equations relating all the unknowns is formed. A least-squares error function is then computed for each candidate, from which an error surface is formed and minima located, providing an estimate of the FOE location. Ego-rotation and inverse depth are then obtained by solving linear equations.

Branca *et al.* [2000] estimate the FOE location by first recovering the camera's egomotion parameters. A sparse displacement field is decomposed into a linear combination of six elementary fields, each corresponding to one of the six 3D motion parameters. A global minimisation is then applied to obtain weights for each elementary field, thus providing solutions for each of the 3D motion parameters, and the location of the FOE. The FOE is used to steer a mobile platform undergoing planar motion, using images from a forward facing camera. The time-to-contact with surfaces projecting onto the FOE is also obtained from the egomotion parameters. In Chapter 4 we discuss the role of FOE estimation for time-to-contact and collision avoidance in more detail.

3.5 Robust depth map recovery

The second task of conventional structure-from-motion based navigation is the inference of egocentric scene structure. Given robust egomotion estimation and a noiseless optical flow field, the scaled depth of projected points in the scene can be directly obtained from Equation 3.12. In practise, however, obtaining accurate depth maps across the field of view is impeded by measurement noise, egomotion estimation error, and incompleteness due to lack of scene texture or feature-points. In addition, ensuring temporal consistency between recovered depth maps poses a significant issue for navigation.

Where depth maps are consolidated or interpreted discretely over time, temporal filtering is typically applied. Matthies *et al.* [1989], for example, apply Kalman filtering to provide predictions of depth for each image location. This prediction is combined with direct depth estimates obtained from optical flow to obtain an overall depth estimate for each image location. Bolles *et al.* [1987] combine feature tracking and the known motion of the camera. More recently, Hung and Ho [1999] apply a Kalman filter using derivatives of the image intensity function to obtain depth maps. At each time step, the known camera translation is used to warp the previous depth map, and obtain new depth values at each point. Results indicate that, with smoothing, reasonable depth estimates can be obtained. Jamal and Venkatesh [2007] estimate depth maps from a proposed colour-based optical flow estimation technique. To account for rotation they maintain an active camera alignment with the direction of motion. Kalman filtering is again applied during depth recovery. The system assumes planar motion, thereby linearising the motion equations.

3.6 Real-time structure-from-motion for navigation

Despite significant work in structure and egomotion recovery, few systems apply such techniques to real-time navigation tasks. By introducing various assumptions, or relaxing the need for the on-line maintenance of dense 3D reconstructions in world coordinate frame, techniques adapting the classical structure-from-motion approach have been applied to mapping and navigation tasks, with the potential for real-time application. We provide an overview of recent work towards this goal below.

3.6.1 Example systems

Nister [2003] proposes a real-time structure-from-motion scheme for estimating camera motion via feature-points under a *Random Sample Consensus* (RANSAC) [Fischler and Bolles 1981] framework. Pre-emptive hypothesis testing is applied prior to the application of RANSAC, thus reducing the sample size, and improving the quality of hypotheses to score. The total number of hypotheses is set *a priori*, allowing them to be computed prior to the application of RANSAC. Nister *et al.* [2004] apply this scheme to the task of general motion estimation (referred to as visual odometry). While it does not explicitly maintain an absolute world model of the environment, a map of the vehicle's travelled path is maintained within a single coordinate frame, making obstacle mapping possible.

Mouragnon *et al.* [2006] apply bundle adjustment incrementally to obtain 3D reconstructions at a significantly faster rate than the classical approach. For each new frame, the camera pose is computed with respect to the previous reconstruction. New points are then matched and reconstructed, after which a local bundle adjustment is applied to refine the model. The method is demonstrated to achieve reasonably accurate localisation estimates over real-world sequences. Whilst the algorithm does not currently support real-time application, such potential is evident.

Assuming camera pose is known, Akbarzadeh *et al.* [2006] report close to real-time performance when obtaining dense reconstructions of urban environments. Data from an on-board global positioning system (GPS) and inertial navigation systems (INS) are fused with a sparse reconstruction from multiple onboard cameras to produce geo-registered 3D maps. A dense reconstruction is then obtained using depth estimates acquired from stereo matching.

Lee *et al.* [2008] propose a scheme for estimating depth from probability distributions of optical flow at specific feature locations. By examining local image intensities, optical flow estimation is accompanied by a Gaussian probability distribution. This distribution is then incorporated into the cost function of a least-squares minimisation scheme to recover egomotion parameters, and the scene depth. The scheme demonstrates adequate robustness to support basic obstacle avoidance over real images acquired from an aerial vehicle. While the approach offers potential real-time application, the computation of probability distributions and minimisation place restrictions on the resolution of the resulting depth maps.

Techniques such as those outlined by Chiuso *et al.* [2000] and Jin *et al.* [2000], do not explicitly apply structure-from-motion solutions to navigation, but offer the potential for real-time performance. They apply nonlinear filtering over feature-point correlations to achieve robust estimates of structure and motion. Chiuso *et al.* [2000] address the specific issue of handling occlusions during the temporal filtering process. The authors report strong performances if sufficient feature-points exist and displacements are small with respect to frame rate.

3.6.2 Summary of structure-from-motion based navigation

There currently exists no structure-from-motion based scheme capable of supporting robust real-time depth map recovery from dense optical flow estimation in the control loop. Existing navigation systems make use of discrete feature-matching to estimate the motion field, and sparse displacement fields to facilitate the accurate and efficient egomotion recovery. These systems do not attempt to recover detailed egocentric depth maps of the environment online. To support general navigation, both accurate egomotion and detailed depth map estimation is necessary.

3.7 Issues for structure-from-motion recovery

3.7.1 Flow field noise

As discussed in Section 3.2, optical flow estimation is inherently ill-posed by virtue of the aperture problem. Thus, noise free image motion estimation is an unrealistic assumption, and one that has impeded the practical application of early solutions to the structure-from-motion problem. While current systems, in general, do not assume noiseless motion estimation, the techniques employed to handle outliers typically require significant computation, and do not necessarily handle the general case. Adiv [1989], for example, notes that under certain conditions, flow field noise introduces inherent ambiguities in structure-from-motion solutions. These conditions include:

- a small field of view,
- fronto-parallel or moderately tilted planar surfaces,
- low translation with respect to depth in the scene,
- a sparse flow field,
- coarse image resolution, and
- noise in adjacent flow vectors being highly correlated.

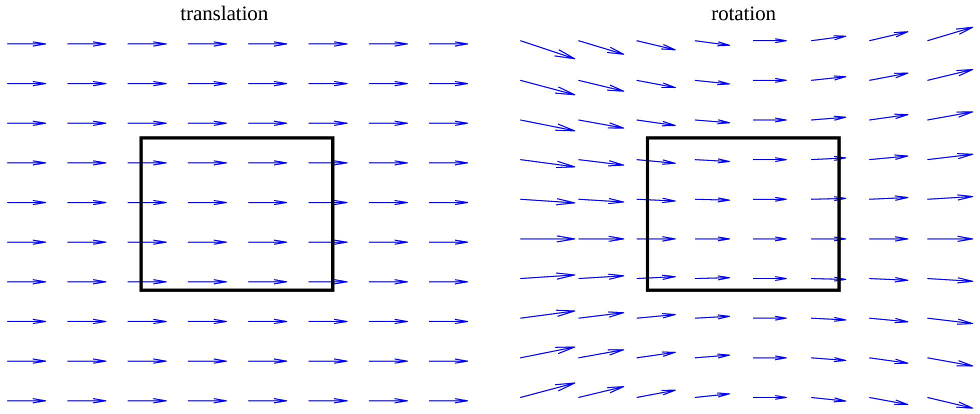

Figure 3.4: A graphical depiction of the translation-rotation ambiguity. The left image shows the flow field resulting from a translation parallel to the X axis; the right image, a rotation about the Y axis. Flow vectors within the central rectangular regions highlight the close resemblance of both flow field patterns when the field of view is small.

Subsequent work by Young and Chellapa [1992] has verified these findings via a statistical analysis of error variances. However, Young and Chellapa show that the application of smoothness constraints during optical flow estimation may reduce the inherent uncertainty in structure-from-motion solutions.

Other studies highlight inherent errors in optical flow estimation itself. Fermüller *et al.* [2001] present an analysis of the statistical bias apparent in local gradient-based estimates of optical flow. Specifically, noise effected gradient estimates are shown to introduce systematic bias that depends on the direction of the flow vector and the distribution of gradient directions and noise. This is shown to be true even in regions where flow is constant, and effects the estimation of both magnitude and direction of the flow vector. Ng and Solo [2001] also note that statistical errors are introduced via finite differencing to obtain image gradient estimates. This is an instance of the *error-in-variable* problem, whereby variables from which a model is obtained, contain errors. Typically, to estimate optical flow from image gradients in local patches, least-squares minimisation is applied. Such techniques assume independence in errors between pixels. In general, however, this is not the case.

3.7.2 Translation-rotation ambiguity

Researchers have noted particular difficulties when the field of view is narrow [Adiv 1989; Verri and Poggio 1989; Daniilidis and Nagel 1993; Fermüller et al. 2001]. This is attributed to an increased coupling between translation and rotation. The ambiguity can be easily visualised when considering the local interpretation of optical flow induced by the translation of the camera in, say, the X direction. Determining whether the resulting horizontal flow vectors are due to a translation in X, or a rotation about Y is difficult (see Figure 3.4). This potential ambiguity is also expressed in the first two constraints of Equations 3.22, where the same value of u_o and v_o can result from a translation in X or Y, or a rotation in Y or X respectively.

For this reason, many structure-from-motion based navigation systems either assume pure translational motion of the sensor (*e.g.,* Chahl and Srinivasan [1997]), or apply planar models to extract surfaces from the scene (*e.g.,* Santos-Victor and Sandini [1997]). While de-rotation algorithms exist, these are largely constrained to a single rotation, or are not fast or robust enough for real-time depth mapping. To generate full 3D depth maps from optical flow, under general motion, an egomotion estimation strategy must solve for all rotational components. To obtain workable depth maps for navigation from flow, the algorithm must be sufficiently accurate, and must provide dense depth map recovery in real-time. Under perspective projection, no such technique has been demonstrated.

3.7.3 Qualitative versus quantitative structure from motion

As noted above, the presence of noise in the optical flow field introduces significant difficulties for uniquely determining structure and motion from optical flow. Based on these observations, Verri and Poggio [1989] argue against the use of optical flow for quantitatively estimating 3D motion and structure parameters. Through an analysis of surface irradiance and its effects on the estimated optical flow field, they show that the optical flow field and the true motion field (*i.e.,* the motion field resulting from the projection of 3D velocities into the image plane) do not, in general, coincide. Indeed, the two align only when a surface with Lambertion reflectance is undergoing pure translation under uniform, fixed illumination. For this reason, Verri and Poggio assert that structure-from-motion methods that rely on local estimates of the optical flow field are unlikely to be accurate.

3.8 Insect-inspired structure-from-motion for navigation

Traditionally, a perspective camera model has been used when inferring scene structure from optical flow. This is in contrast to insect vision, where the compound eye structure of most insects provides an almost global view of the scene [Chahl and Srinivasan 1997]. There is a growing body of theoretical work suggesting a near global field of view, often described by a spherical projection model, may offer distinct advantages when inferring scene structure and self-motion from optical flow [Fermüller and Aloimonos 2000]. Below, we overview arguments and motivate consideration of a spherical projection model for structure-from-motion recovery for navigation.

3.8.1 Geometric advantages of spherical projection

Geometric properties of the sphere have been shown to facilitate more efficient and robust interpretations of optical flow. For example, Brodsky *et al.* [1998] show that on a full view sphere, optical flow can be unambiguously interpreted on the basis of the direction of flow vectors alone. Given a hemispherical projection, two different rigid translations and rotations cannot induce the same motion field on the sphere unless the plane of the translational vectors of both motions is perpendicular to the plane of rotational vectors of both motions. Allowing the use of flow magnitude, however, provides an unambiguous interpretation of the flow field given a hemispherical view.

Fermüller and Aloimonos [1998] show that depth maps generated under a spherical projection model are inherently more stable than under perspective projection. Stability is measured on the satisfaction of positive depth across the field of view. Under perspective projection, depth map distortion introduced via errors in estimated egomotion parameters is shown to be dependent on the relative orientation of the translational and rotational axes. In contrast, depth map distortions resulting from the same error under spherical projection are not effected by their relative orientation. Thus, the best achievable depth map under spherical projection is obtained by assuming de-rotation is correct.

3.8.2 Estimating egomotion from spherical optical flow

The view sphere has been identified as providing particular advantages for the estimation of egomotion. In early theoretical work, Nelson and Aloimonos [1988] highlight three key geometric properties of optical flow on the sphere that may be exploited to recover egomotion:

1. the component of flow parallel to any great circle is effected only by the rotational component about its perpendicular axis, thus decoupling it from rotations about orthogonal axes.

2. under pure translation, both the FOE and the focus of contraction (FOC) will be located at antipodal points on the sphere, and will evenly partition flow along any great circle connecting these two points, into two distinct directions of motion (*i.e.,* clockwise and counter-clockwise).

3. the existence of any rotational motion along a great circle causes the FOE and FOC to converge, thus ensuring the two points will only lie at antipodal locations under pure translation.

The first observation indicates that each component of rotation can be resolved independently, and thus each may be considered in turn. Observations 2 and 3 suggest a simple search strategy over a range of possible rotations can be employed to recover the rotation about the axis of the great circle under consideration. After de-rotation, the direction of translation is given by the line passing through the FOE and FOC. While simulation and some theoretical analysis of the algorithm's performance is given, no published results report this algorithm's application to real-time navigation tasks, over real image sequences. In this thesis, we consider the possible application of this algorithm for the generation of real-time 3D depth maps in real-time to support structure-from-motion for navigation (Chapter 7).

Recently, Lim and Barnes [2007, 2008] have proposed techniques for recovering egomotion on the view sphere, by examining the direction of optical flow vectors at antipodal points on the view sphere. Examination of the direction of flow vectors at antipodal points provides a constraint on the possible directions of camera motion. Taking many such antipodal flow vector samples provides further constraints on this sub region, leading to a consensus-based estimate of egomotion. The key advantage of this approach is that it avoids searching the motion parameter space, and provides increased robustness to erroneous flow estimates. A practical drawback of the technique, however, is the explicit requirement for correspondences of flow vectors at antipodal points within the image. This may be problematic in environments where significant portions of the scene are featureless (*e.g.,* flying above a planar surface or moving alongside a featureless wall).

3.8.3 Summary of insect-inspired structure-from-motion

Despite strong theoretical justification for the use of a spherical projection model in structure-from-motion recovery, practical applications of the approach are only beginning to be explored. Recent examples include Maddern and Wyeth [2008], who describe the design of a hemispherical compound optical flow sensor to support robust 3D egomotion on a miniature aerial vehicle. Dengate *et al.* [2008] consider the use of a hemispherical camera for self-motion recovery in a wearable low-vision assistive device. There remains, however, a need for further experimental validation of the theoretically identified geometric and computational advantages of a spherical projection model for structure-from-motion. In addition, there is a need for consideration of potential gains under other systems of flow-based visual navigation such as visuo-motor control.

3.9 Summary

In this chapter we have reviewed work in the estimation of optical flow and the interpretation of scene structure and self-motion from the optical flow field. Specifically, we have derived the differential invariants of the local affine flow field: `div`, `def` and `curl`, and their respective relationships with local motion and structure in the scene. These

Table 3.1: Summary of Structure-from-motion (SFM) techniques discussed in this chapter.

SFM Solution	Data	References
Closed-loop local SFM	Optical flow + differentials	Koenderink and Van Doorn [1975][1976], Longuet-Higgins [1984] and Prazdny [1980], Waxman and Ullman [1985], Kanatani [1987], Subbarao [1989][1990], Cipolla and Blake [1997]
	Feature points	Tsai and Huang [1981], Longuet-Higgins [1981]
Egomotion	Optical flow field	Bruss and Horn [1983], Adiv [1985], Heeger and Jepson [1992], Chiuso *et al.* [2000], Oliensis [2005]
	Optical flow (FOE)	Jain [1983], Rieger and Lawton [1985], Negahdaripour and Horn [1989], Li [1992], Sazbon *et al.* [2004], Joarder and Raviv [1994]
	Optical flow (FOE + depth constraint)	Negahdaripour [1996], McQuirk *et al.* [1998], Srinivasan *et al.* [1999][2000], Branca *et al.* [2000]
	Omni-visual flow	Nelson and aloimonos [1988], Lim and Barnes [2007][2008], Maddern and Wyeth [2008], Dengate *et al.* [2008]
Depth map (given egomotion)	flow + filtering	Matthies *et al.* [1989], Hung and Ho [1999], Jamal and Venkatesh [2007]
	1D Omni-visual flow	Chahl and Srinivasan [1997]
	Feature points + filtering	Bolles *et al.* [1987]
SFM for navigation	Sparse optical flow	Lee *et al.* [2008]
	Feature points	Nister [2003][2004], Chiuso *et al.* [2000], Jin *et al.* [2000], Mouragnon *et al.* [2006]
	Feature points + INS/GPS	Pollefeys *et al.* [2008]
	Full omni-visual flow	**Chapter 7**

relationships provide the foundations of the major contributions of this thesis.

We have considered navigation in the context of structure-from-motion, via the explicit estimation of egomotion and scene structure parameters. Table 3.1 provides a summary of all the structure-from-motion techniques discussed in this chapter. Despite continuing improvements in structure-from-motion techniques for real-time applications, there currently exist no reported work demonstrating full structure-from-motion recovery from dense optical flow capable of supporting navigation in the control loop. Chapter 7 of this thesis (referred to in last row of Table 3.1) addresses this need by proposing a new structure-from-motion scheme using dense optical flow under a spherical projection model. To this end, we have reviewed theoretical arguments in favour of a spherical projection model.

In the next chapter we consider the application of optical flow in the control loop under a visuo-motor control framework. In particular, we motivate and examine the use of time-to-contact as an important visual quantity for visuo-motor control.

Chapter 4

Time-to-contact for visuo-motor control

4.1 Introduction

We have now motivated the use of optical flow for visual contact estimation and explored techniques for inferring scene structure and egomotion from the optical flow field. In particular, we considered visual navigation using general solutions to the structure-from-motion problem, highlighting inherent issues impeding the general application of the methodology to real-time navigation tasks. As an alternative approach to visual contact estimation, we now consider the use of optical flow for inferring time-to-contact. In particular, we focus on the role of time-to-contact in supporting visuo-motor navigation tasks such as collision avoidance, docking and landing.

The contributions of this thesis are primarily concerned with the robust estimation and use of time-to-contact as an input to visuo-motor control. In this chapter we review the underlying theory of time-to-contact estimation, and previous work demonstrating its application to vision-guided navigation. We focus specifically on tasks requiring fine motion control such as landing and docking, for which the demands on robust estimates of time-to-contact are high. We argue that current time-to-contact estimation techniques do not provide sufficient robustness, or generality of application, to support such tasks.

The chapter is structured as follows. Section 4.2 defines and motivates the use of time-to-contact over traditional structure-from-motion solutions. Section 4.3 provides an explicit derivation of time-to-contact from local flow field differential invariants.

Section 4.4 discusses techniques for estimating time-to-contact before discussing its application to collision avoidance in Section 4.5, and controlled surface approaches (*i.e.,* docking and landing) in Section 4.6. Section 4.7 discusses current issues in estimating and applying time-to-contact to such tasks. A chapter summary is provided in Section 4.8.

4.2 Time-to-contact

4.2.1 Definition and motivation

Time-to-contact (often referred to as τ) is defined as the ratio of surface distance to velocity towards the surface [Lee 1976]. Thus, for a pinhole camera translating along its optical axis, the time-to-contact is defined as:

$$\tau = \frac{Z_o}{T_z}, \tag{4.1}$$

where Z_o is the distance to the surface, and T_z is the velocity towards the surface (or analogously, the surface towards the camera).

A key motivation for computing time-to-contact is that it may be recovered without the need for solving the complete structure-from-motion problem. Rather, time-to-contact is a directly measurable cue from local differential invariants of optical flow. In contrast to the scaled depth of points in the scene, time-to-contact shifts the unit of measure from a spatial metric to a temporal one. Thus, surface distances are measured in terms of the time it would take to collide with that surface, given the instantaneous component of the observer's velocity in that direction.

Measuring proximity in this way has important implications for motion control design. Most significantly, the cue directly relates the observer's motion to the scene structure without any requirement for the explicit recovery of the observer's direction of heading. In addition, measuring proximity in temporal units provides a more intuitive means in which to make control adjustments such as the speed of approach. A temporal unit of distance measure also satisfies the need for a predictive cue upon which to base motor control adjustments.

4.2.2 Biological support

There is strong evidence in support of time-to-contact as a fundamental visual cue for motor control across a wide range of animal species. Studies of locusts and other flying insects have highlighted selective neural responses to looming visual stimuli enabling evasive actions to avoid collision [Rind 1997; Robertson and Johnson 1993]. Neural mechanisms have also been identified to control motor actions during approaches towards surfaces in flies [Wagner 1982], and in birds such as gannets [Lee and Reddish 1981], and pigeons [Lee et al. 1993]. In all cases, the neural response has been shown to directly correspond with visual looming. How time-to-contact is perceived in primate vision is less understood, though the underlying sensory cues and associated neural mechanisms have been given considerable attention [Lappe 2004].

4.3 Inferring time-to-contact from optical flow

Subbarao [1990] was the first to provide an explicit derivation of time-to-contact from local flow field differential invariants. For completeness, we provide an overview of the derivation here.

Recall the decomposition of a local optical flow patch into the differential invariants: `div`, `curl` and `def`, defined in Equations 3.25. Recall also, Equations 3.32-3.35, which express these invariants in terms of the 3D motion and structure of a piece-wise planar surface projecting to a small patch about the image origin.

We re-define the scaled parallel components of translational motion such that:

$$\frac{T_x}{Z_o} = m\cos(\psi) \qquad \frac{T_y}{Z_o} = m\sin(\psi), \tag{4.2}$$

where m is the signed magnitude of parallel translation, and ψ is the direction of the parallel translation in the image plane. Similarly, we redefine the direction of the depth gradient of the planar surface such that:

$$a = f\cos(\phi) \qquad b = f\sin(\phi), \tag{4.3}$$

where ϕ is the direction of the depth gradient parallel to the image plane, and f is the signed magnitude of the depth gradient. Substituting the above into Equation 3.32, we obtain:

$$\begin{aligned} \texttt{div} &= \frac{2T_z}{Z_o} + mf\Big(\cos\psi\cos\phi + \sin\psi\sin\phi\Big) \\ &= \frac{2T_z}{Z_o} + mf\cos(\psi - \phi). \end{aligned} \tag{4.4}$$

With simple algebraic manipulation, we obtain the following equation:

$$\frac{T_z}{Z_o} = -\frac{1}{2}\Big[\texttt{div} + mf\cos(\psi - \phi)\Big], \tag{4.5}$$

thus defining the recipricol of the time-to-contact.

The above highlights an example of the deformation induced ambiguity when inferring structure and motion quantities from the divergence. Without knowledge of either the surface gradient direction in the image (ϕ), or the direction of translational motion in the image (ψ), it is not possible to calculate this term precisely. Noting, however, that $-1 \leq \cos(\psi - \phi) \leq 1$, we may define τ as a bound, such that [Subbarao 1990]:

$$\frac{1}{2}\Big(\texttt{div} - \texttt{def}\Big) \leq \tau^{-1} \leq \frac{1}{2}\Big(\texttt{div} + \texttt{def}\Big). \tag{4.6}$$

Given `div` and `def` are both defined in terms of first-order partial derivatives of flow, we may define the bound on τ in terms of these measurable visual quantities [Subbarao 1990]:

$$\frac{1}{2}\Big(u_x + v_y - \sqrt{(u_y + v_x)^2 + (u_x - v_y)^2}\Big) \leq \tau^{-1} \leq \frac{1}{2}\Big(u_x + v_y + \sqrt{(u_y + v_x)^2 + (u_x - v_y)^2}\Big). \tag{4.7}$$

4.3.1 Generalising time-to-contact to any viewing direction

In this thesis, we consider time-to-contact estimation at locations across entire viewing areas, and under both a perspective and spherical projection model. Equation 4.7, however, defines the bound on time-to-contact along the viewing direction only. Thus, given an image under perspective projection, the bound is only valid along the optical axis. To consider time-to-contact estimation along different visual angles, a generalised definition of time-to-contact (Equation 4.1) is required.

Addressing this issue, Colombo [1999, 2000] shows that a bound for time-to-contact can always be computed, regardless of the field of view, if time-to-contact is redefined to encompass the non-decoupled *total relative velocity* of the surface and camera. Specifically, if $\dot{P}$ is the total velocity of a point P, such that $\dot{P} = T_p + \Omega_p$, Colombo redefines time-to-contact under perspective projection as:

$$\tau_p = \frac{P_z}{|\dot{P}| \cos \beta}, \tag{4.8}$$

where β is the angle between the visual ray passing through P, and the optical axis, and P_z is the depth of the point in the direction parallel to the optical axis. In addition, Colombo defines time-to-contact under a spherical projection model as:

$$\tau_s = \frac{|P|}{\dot{P} \cdot r}, \tag{4.9}$$

where $|P|$ is the radial distance to a surface point projecting to a location, r, on the image sphere. τ_s provides a radial definition of time-to-contact. Note that computing τ_p within the tangent plane about r yields the same definition of time-to-contact. This yields two important implications: (i), a bound on τ_s will always exist, and is defined as in Equation 4.6 within the local tangent plane; and (ii), by mapping this bound to a tangent perspective image plane to the sphere, a bound for τ_p can also be computed for any viewing direction.

Another significant contribution of these definitions is that a clear distinction between time-to-contact and scaled depth is provided. While time-to-contact and scaled depth are the same along the optical axis, these quantities diverge rapidly for points away from the image centre.

4.4 Computing time-to-contact

Numerous approaches have been adopted for computing time-to-contact. We review the major classes of techniques applied below.

4.4.1 Time-to-contact from flow field models and approximations

To avoid the overheads of computing full optical flow fields, many early time-to-contact estimation techniques made use of flow approximation models. Most commonly, these approximations are used to generate motion models for surfaces in the scene, from which time-to-contact can be estimated.

Meyer [1994] assumes planar motion to obtain first order image motion parameters for a smooth, rigid surface in relative motion with respect to the camera. From this, an equation is derived for estimating time-to-contact for any point in the image in terms of the first-order coefficients of the flow field and the surface orientation parameters. Meyer notes that the surface orientation parameters cannot be recovered given only forward translational motion of the camera. To obtain the first-order flow coefficients, a multi-resolution scheme is used to refine the estimate of the six parameters. Considering the constraint for every point in the local region provides an over-constrained system of linear equations, which can then be solved using least squares. The proposed method assumes the vertical component of the motion field is zero. This assumption, however, will only hold if the camera's optical axis is parallel to the ground plane.

Assuming a flat ground plane, Santos-Victor and Sandini [1996] apply a model-based approach to estimate the range of surfaces lying on the ground plane. Using knowledge of the geometrical arrangement of a camera at a fixed and calibrated angle with respect to the ground plane, obstacles are detected as regions of the image where the estimated ground plane motion model is violated. This is demonstrated through a a simple obstacle avoidance strategy for a mobile robot.

Applying a similar approach, Lourakis and Orphanoudakis [1999] propose a technique for computing time-to-contact for obstacles on the ground plane. The technique estimates a motion model for the ground plane from the optical flow, and then subtracts the ground plane motion from the estimated optical flow, yielding regions where obstacles exist. Using the motion model, time-to-contact is computed for points on the ground plane. Planar parallax is then employed to estimate time-to-contact for points off the ground plane. The technique is demonstrated to robustly estimate time-to-contact with obstacles across a real image sequence. The technique assumes a planar ground, and requires that this surface remain sufficiently visible in the image. Thus, the technique is only valid for collision avoidance tasks.

Arnspang *et al.* [1995] combine estimates of optic acceleration with the normal flow field to estimate time-to-contact along curve segments in the image. The use of curves circumvents the aperture problem, allowing use of the normal flow field.

The most significant drawback of model-based time-to-contact estimation is the requirement for recovering structure-from-motion parameters in order to solve for time-to-contact. Such techniques typically impose restrictions on camera motion to avoid recovering full structure-from-motion solutions.

4.4.2 Time-to-contact from closed-contours

4.4.2.1 Green's theorem

Time-to-contact can be estimated by examining the temporal changes in the moments of area of a closed-contour. Such techniques are based on Green's theorem [Kaplan 1991], which provides a means of estimating the divergence of a region without the explicit computation of optical flow.

Given a projected surface patch, $S \in \mathbb{R}^3$, whose image is bounded by a closed contour, $C \in \mathbb{R}^2$, and with a flow field, $\vec{\mathbf{U}} \in \mathbb{R}^2$, defined continuously across S, Green's theorem states that the integral of flow vectors along C in the direction of the normal to C, is equal to the integral of the divergence of flow vectors defined on S. That is, the average divergence of the surface patch S can be obtained by summing normal vectors along C. This can be expressed as:

$$\int\int_S \mathtt{div}\ \vec{\mathbf{U}}\ ds = \oint_C \vec{\mathbf{U}} \centerdot \vec{n}\ dl, \tag{4.10}$$

where $\vec{\mathbf{n}}$ is a component of flow normal to C, ds is an element of S and dl is an element of C [Duric et al. 1999]. The average time-to-contact for the surface patch is simply the recipricol of this summation [Duric et al. 1999]. Figure 4.1 shows this graphically.

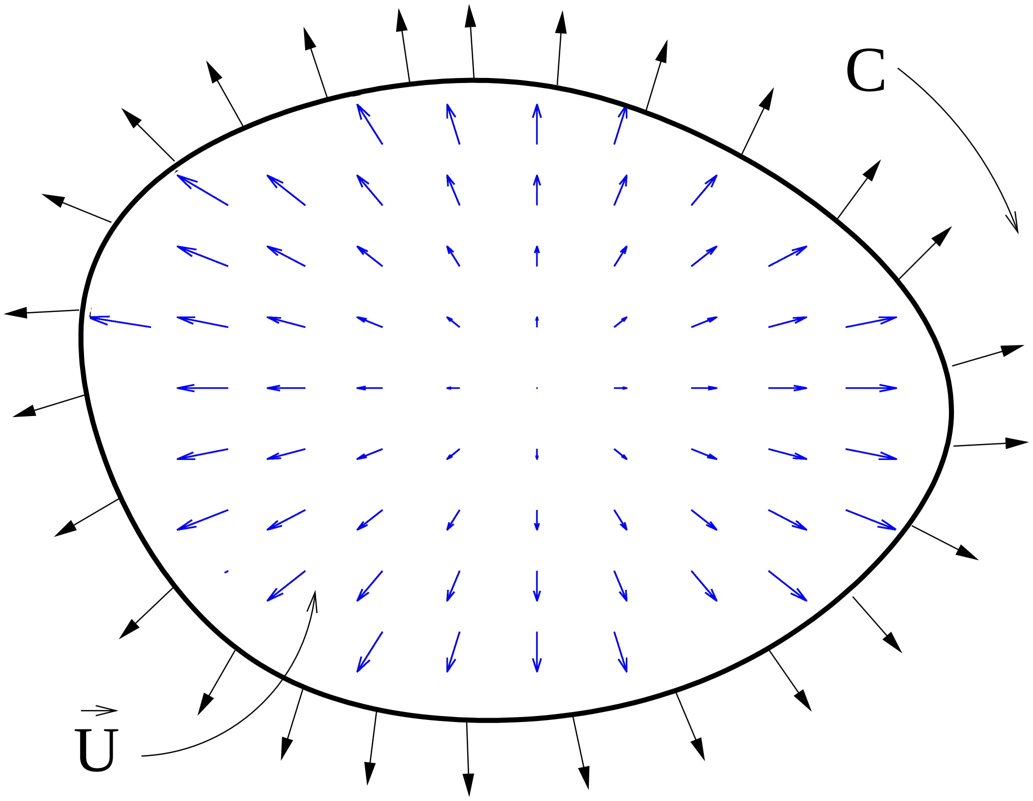

Figure 4.1: Green's theorem applied to the computation of divergence. The average divergence of the optical flow, $\vec{\mathbf{U}}$, within the closed contour, C, is given by the integral of the components of flow normal, $\vec{\mathbf{n}}$ to C, along C.

4.4.2.2 Closed-contour examples

Maybank [1987] was the first to apply Green's theorem to the estimation of the rate-of-approach. He shows that for a small image patch, the rate of change of the apparent area of a rigidly moving object can be expressed as an integral, from which an approximation to the time-to-contact of the object can be obtained if the axis of motion passes through the patch. The same approximation is obtained using the divergence of the flow field within the patch.

Cipolla and Blake [1997] propose a technique for estimating surface orientation and time-to-contact by examining temporal changes in the moments of area of a closed-contour. Closed-contours are tracked via B-spline snake control points, which after an initial radial search from the image centre to find contour points, are then tracked via the image motion. While local affine approximations of the flow field do not provide sufficient constraints upon which to recover full motion and structure, Cipolla and Blake augment this with additional constraints. Specifically, they assume that the direction of translation is deliberate, and therefore known. This assumption resolves the planar deformation ambiguity. Experiments demonstrate the application of the time-to-contact estimation scheme for computing the proximity of approaching objects, performing a controlled approach to a surface, using the divergence for time-to-contact estimation, and the deformation to align the camera fronto-parallel with the surface.

Duric *et al.* [1999] consider the estimation of the average rate of approach (*i.e.,* τ^{-1}) for a closed surface patch on the view sphere, from the expansion of the patch. Using Green's theorem, they derive an equation relating the average τ^{-1} of a surface patch in the scene to the orientation of the surface patch, and the integral of the normal motion field along the boundary of the spherically projected patch. From this, a desired upper bound on the rate of approach of the patch with respect to the area of the closed patch is derived.

Di Marco *et al.* [2003] propose a set-theoretic approach to closed-contour time-to-contact estimation. Temporal changes of the contour region are tracked via the recursive application of a set membership filter. At each time step, the filter computes a set of state vectors (*i.e.,* affine transformation parameters of the region, and their time derivatives) consistent with current measurements and previously computed state approximations. Time-to-contact is then computed from a central estimate of possible state vectors. The spread of possible state vectors gives a direct indication of uncertainty in the estimate. Results indicate some sensitivity to initial value choices, however, the algorithm is capable of detecting when time-to-contact estimates fall outside the error bounds.

Closed-contour time-to-contact techniques are generally motivated by a desire to avoid computing optical flow explicitly. It is important to note, however, that Green's theorem provides only an affine approximation to the projected surface deformation. In small regions, affine models provide an adequate approximation of non-planar surfaces. However, such models are likely to break down as the surface draws closer, particularly if the approach is non fronto-parallel with the surface tangent plane. An additional issue is finding a closed contour to track when in close proximity with a surface. If the field of view is restricted, then closed-contour regions are likely to grow larger than the image plane. Thus, closed-contour time-to-contact estimation may be ill-suited to tasks requiring fine motion control in close proximity with a surface.

4.4.3 Time-to-contact from space-variant maps/sensors

Light receptors of the human retina are not uniformly distributed. Rather, they are at highest density about the fovea (effectively the eye's optical axis), and decreasingly so at locations radially away from this point [Tistarelli and Sandini 1993]. Researchers have attempted to mimic this topology via the use of space-variant sensors, or mappings, where the sampling distance between pixels is linearly increased away from the projective centre. A common representation of the mapping is on a Cartesian plane, where the dimensions represent both components of the polar coordinates of each pixel. Thus, one dimension represents the radial position of a pixel with respect to the fovea, and the other, its angular position on a circle centred on the fovea.

Tistarelli and Sandini [1993] argue that considering optical flow under such a representation holds particular advantages for the estimation of time-to-contact. Specifically, they show that a direct estimate of the time-to-contact can be obtained for any surface orientation (*i.e.,* not just the bounds) by considering the partial derivatives of flow in the radial direction of the mapping. It is shown that only the radial component of motion relates to the time-to-contact, and is equivalent to estimating the flow field divergence. Notably, however, their derivation of time-to-contact assumes the surface gradient is locally zero. This effectively assumes the surface is fronto-parallel at each imaged point, and thus cannot provide a precise time-to-contact estimate for inclined surfaces away from the projective centre [Colombo 2000].

It is important to note that while such foveated representations do provide particular advantages, these are typically of more relevance to fixation-based approaches. The inherent assumption of this representation is that a single point in the field of view represents the point of interest, or at least, that there is some capability of fixating on such a point. Where points of interest are potentially numerous, more globally constant representations are likely to be more appropriate.

4.5 Time-to-contact for collision avoidance

The primary use of time-to-contact in robot navigation has been to facilitate obstacle avoidance. We review existing techniques for achieving this below. We limit our review to techniques that estimate time-to-contact directly from visual motion information.

Nelson and Aloimonos [1989] were first to implement a simple obstacle avoidance algorithm using flow divergence for a camera mounted on a robot arm. The camera is guided between two obstacles by orienting motion towards areas of minimal flow divergence. The motion of the sensor is not continuous, having to pause before each directional update.

Ancona and Poggio [1993] propose a simple 1D correlation-based approximation method for estimating the elementary motion components of a linear optical flow field. Assuming translational motion only, they estimate time-to-contact with a looming surface using 1D correlation patches placed symmetrically about a circle centred on the image origin. Noting the invariance of divergence to the location of the FOE under an affine flow model, Ancona and Poggio exploit Green's theorem, treating the circle of correlation patches as a closed-contour. Summing the radial component of estimated flow from each patch, they obtain the divergence, and hence, the time-to-contact. The scheme is demonstrated over a sequence obtained from a mobile platform undergoing constant forward translation only.

Coombs *et al.* [1998] use normal flow in the central region of the camera view to recover flow divergence for real-time time-to-contact estimation. Time-to-contact is used to decide whether to turn or to stop when collision was imminent.

More recently, researchers have considered flow-based collision avoidance strategies for non-ground-based robots. In particular, focus has been given to insect-inspired visuo-motor control schemes based on the neural circuitry of insect vision. Much of this work employs hardware-based Elementary Motion Detectors (EMDs) to estimate visual motion. EMDs are based on the motion detection of insects [Reichardt 1969].

Zufferey and Floreano [2006] propose a divergence-based obstacle detector for invoking evasive steering responses using EMD-based flow estimation. Divergence is measured along the optical axis of a forward-facing camera by taking the difference of flow in the left and right of the image. A steering response is invoked to avoid collision if divergence exceeds a preset threshold. The direction of steering is determined by the relative balance of flow magnitudes in the left and right views, however steering itself is not controlled using visual inputs.

A similar saccading strategy for obstacle avoidance is demonstrated by Green *et al.* [2004] on a micro air vehicle. Unlike Zufferey and Floreano, they do not use the divergence of flow, and instead base obstacle detection on translational flow in the periphery of the frontal view. Thus, they assume translational motion only (or flow field de-rotation).

Bermùdez *et al.* [2007] demonstrate a locust-inspired collision detection scheme for a flying robot. The detection method is modelled on the Lobula Giant Movement Detector (LGMD) in locusts. EMD responses are integrated in the LGMD model, from which an output spike is produced when visual motion is divergent. LGMD responses are also integrated over time. A collision detection is triggered if the value exceeds a pre-set threshold.

4.5.1 Summary of collision avoidance systems

For time-to-contact based obstacle avoidance, the primary task is the detection of looming obstacles. The resulting evasive response, however, is typically performed in open-loop over a preset time duration, for which no continuous use of visual quantities such as time-to-contact is required. Thus, the demands on high accuracy and temporal consistency in proximity estimates are relatively low. For this reason, time-to-contact estimation has been most successfully applied to collision avoidance in robot navigation. Where continuous use of time-to-contact is required for fine motion control, particularly in close proximity with surfaces, estimation strategies such as those outlined above are unlikely to provide sufficient robustness.

4.6 Time-to-contact for docking and landing

We now review current techniques for performing controlled approaches to surfaces using visual motion as the primary cue for velocity and alignment control. Note that we do not include landmark-based visual servoing strategies for docking such as those in Wei *et al.* [2005] and Usher *et al.* [2003], which base velocity and pose control on the location of scene features in the image. Rather, we focus on techniques that derive control schemes using the optical flow field.

4.6.1 The docking problem

Arkin and Murphy [1990] break the general docking problem into two phases: *ballistic* and *controlled*. The ballistic phase seeks to quickly navigate the robot to the general target area. The controlled phase then employs the fine direction control to accurately position the robot in preparation for the final deceleration (and any other interaction with the docking surface). When reaching the controlled phase in Arkin and Murphy's model, two operations must occur:

- *alignment*: the robot seeks to minimise an angular error between its current heading and the target orientation. When docking with surfaces, the target orientation is most often a fronto-parallel alignment with the surface.
- *braking*: the robot decelerates to a halt as close as possible to the desired location (often the docking surface).

It is important to note that these tasks are not strictly sequential. However, it is often a requirement that alignment with a docking surface be achieved before fine velocity control becomes crucial [Questa et al. 1995]. One common reason for this is that time-to-contact estimates obtained from flow field measures such as divergence, typically assume a fronto-parallel orientation with the surface.

4.6.2 Docking and landing systems

Santos Victor and Sandini [1997] align the docking surface through the use of parameters obtained from an affine approximation to the optical flow field. Normal flow is measured and an affine model is applied to obtain the approximated flow field. Affine parameters allow the surface normal to be identified. The control scheme then seeks to minimise the angle between the surface normal and the optical axis of the camera.

Forward velocity is also controlled via affine parameters of the approximated motion field. They present two forms of docking:

- *ego-docking*: visual data is acquired from a camera mounted on the robot performing the docking behaviour, and used to guide the manoeuvre.
- *eco-docking*: the camera and computation resources are located at the docking surface and used to guide the robot to the surface.

Questa *et al.* [1995] use normal flow to approximate the affine parameters of the flow field. From this, the divergence is obtained and used to regulate the velocity of a robot arm with mounted camera attempting to dock with a fronto-parallel planar surface. Fronto-parallel alignment is achieved via a combination of lateral translational adjustments in the direction of increasing surface depth, and opposing rotations of the camera to maintain fixation on a point on the surface. The surface depth gradient is obtained by substituting the known direction of parallel motion of the camera into Equation 4.2. A similar alignment strategy is applied by Santos-Victor and Sandini [1997]. The time-to-contact estimate obtained from the divergence is then used to reduce velocity as the end effector approaches the docking surface. This is achieved by reducing forward velocity in inverse proportion to increasing flow divergence.

Results reported by Questa *et al.* [1995] exhibit some sensitivity to errors in alignment with the docking surface. The docking strategy should, in theory, achieve fronto-parallel alignment with the docking surface before fine motion control is required. However, average angular errors in alignment of 8^o were recorded in the final stages of the docking manoeuvre, giving rise to errors in the time-to-contact estimate. Significant oscillation in the alignment error is also reported at close proximity.

Similar work by the same laboratory [Questa and Sandini 1996] has also demonstrated the use of a space-variant log-polar sensor to achieve fronto-parallel docking manoeuvres using a camera-mounted end-effector. Time-to-contact is estimated from the component of image motion in the radial direction. (*i.e.,* the divergence resulting from motion along the viewing direction). To achieve docking, pure translation along the viewing direction is assumed. While not implemented on-board, Barnes and Sandini [2000] provide a mathematical formulation for the use of the rotational component of a log-polar image representation to achieve directional control in docking.

Issues with precise docking have been addressed by Mandel and Duffie [1987]. They account for errors in positioning of a robot manipulator, thereby allowing precise interactions with some allowance for imprecise docking.

In recent years, there has been interest in the use of optical flow for visuo-motor control of in-flight landing control systems. In particular, researchers have attempted to apply insect-inspired strategies to perform safe, repeatable landing manoeuvres using the apparent motion of the landing surface.

Srinivasan *et al.* [2000] demonstrate the honeybee graze-landing model (described in Section 2.6.4) on a robot gantry. Forward speed is reduced by holding the angular motion of the ground plane constant, while at the same time reducing the speed of descent proportionally. In [Chahl et al. 2004], the system is demonstrated to maintain a preset angle of descent onboard a fixed-wing model aircraft. In both cases, the camera is pointed vertically down towards the ground plane, and is assumed to remain at this alignment during the approach.

Ruffier and Francheschini [2005] apply a similar landing model on a rotor-craft. Optical flow is estimated via an EMD, the outputs of which are fed directly into visuo-motor control schemes. Landing is achieved by tilting the rotor-craft's nose to decrease forward velocity. The flow-based controller then reduces the height of the vehicle so as to maintain constant horizontal flow. The system is demonstrated to produce repeatable graze-landing approaches, and to operate in wind effected conditions. Green *et al.* [2004] also demonstrate the graze-landing model proposed by Srinivasan *et al.* [2000] using a downward pointing camera onboard an ultra-light fixed-wing model aircraft.

4.7 Issues for time-to-contact based visuo-motor control

4.7.1 Flow field approximations

An important drawback of many of these approaches is the requirement for the explicit segmentation of a surface in order to estimate the image motion. Where closed-contour deformation is measured, there is also the problem of reliably finding closed shapes when in close proximity with the surface [Cipolla and Blake 1997]. EMD-based motion estimation, while fast, is sensitive to image contrast [Borst and Egelhaaf 1993]. As such, the amplitude of motion detected will be greater where higher contrast exists given the same underlying motion. It is therefore incapable of providing precise visual motion estimates.

An alternative approach is to compute time-to-contact from general optical flow. Methods for estimating general optical flow fields from local image regions, such as Lucas and Kanade's method [1981], require no *a priori* knowledge of scene structure, and therefore, no segmentation. In general, for systems such as road vehicles, optical flow is often used for other functions, such as a general sensor for salience to detect moving hazards over the whole scene, as well as for particular functions such as obstacle detection. Affine approximations of image motion are not adequate for this type of general use, and having multiple methods for calculating flow is implausible on restricted embedded hardware. There currently exists no docking or landing scheme based on time-to-contact estimates taken from full, general optical flow.

4.7.2 Robustness during egomotion

Mobile robot ego-motion is rarely precise, and even where only translational motion is intended, rotations will be present. Small directional control adjustments, fluctuations in direction due to steering control or differing motor outputs, bumps and undulations along the ground surface, and noisy optical flow estimation will all cause instantaneous, frame-to-frame rotations of the robot. This subjects the optical axis to small rotations about the predominant direction of motion. As a result, the FOE is unlikely to be fixed with respect to the image centre. In previous work with divergence-based time-to-contact estimation, divergence is almost always measured at the same image location in each frame [Ancona and Poggio 1993; Nelson and Alloimonos 1989; Coombs et al. 1998]. Ancona and Poggio, for example, use simple linear motion detectors to estimate flow in orthogonal directions at locations symmetrically placed about the image centre. This, however, ignores the effect of FOE shifts on the divergence measure across the image.

Previous work has addressed aspects of this issue. Subbarao [1990] considers time-to-contact with surfaces of arbitrary orientation, for a camera of arbitrary alignment with respect to the direction of motion. Subbarao, however, assumes the point of interest lies along the camera's optical axis. While a fixation-based strategy such as that used by Questa *et al.* [1995] can keep the target point centred, a mobile robot is unable to achieve this without additional hardware support. In many cases, such hardware is unavailable to facilitate high speed fixation.

An alternative approach is to account for instantaneous rotations in the image domain, by tracking the location of the FOE. Van Leeuwen and Groen [2002, 2000] consider the use of FOE tracking to correct for the physical misalignment of the optical and translational axes as a result of the camera-robot configuration. However, while accounting for the constant physical misalignment of these axes, they do not extend the use of FOE tracking explicitly to the removal of small frame-to-frame rotational effects during ego-motion, nor do they apply time-to-contact directly to control the vehicle's velocity. In summary, while previous work has considered the use of FOE tracking for camera stabilisation during ego-motion, no one has applied such an approach to tasks requiring fine motion control (such as docking), nor provided a theoretical analysis supporting the advantages of such a strategy, and its potential use for control.

4.7.3 Surface orientation and alignment

There currently exists no time-to-contact docking/landing strategy capable of performing controlled approaches to surfaces of arbitrary orientation. While systems such as Santos-Victor and Sandini [1997] facilitate directional adjustments to achieve fronto-parallel alignment with a docking surface, they do not support docking at non-frontal angles.

A primary reason for fronto-parallel alignment is the need for accurate time-to-contact estimates when in close proximity with the surface. As the approach angle moves away from fronto-parallel, the bounds on time-to-contact increase due to increased deformation of the projected surface. If motion cannot be assumed to be along the optical axis (and in general, it cannot), then these techniques are effectively restricted to fronto-parallel approaches, and time-to-contact must be measured close to the image origin.

While Srinivasan *et al.* [2000] propose a scheme for performing graze landings

Table 4.1: Summary of time-to-contact estimation scheme assumptions.

Approximation	Measurement location	Reference
Narrow FOV	Image origin	Subbarao [1990], Nelson and Aloimonos [1989], Ancona and Poggio [1993]
Known/recovered surface orientation	Any image point	Meyer [1994], Santos-Victor and Sandini [1996], Srinivasan *et al.* [2000]
Affine model over segmented surface	Any image point	Maybank [1987], Cipolla and Blake [1997], Duric *et al.* [1999], Di Marco *et al.* [2003]
Planar surface	Any image point	Chapter 6

(*i.e.,* non-fronto-parallel approaches), two issues constrain its general application:

1. the estimation of time-to-contact is obtained from the first-order image motion of the ground plane, and thus the model assumes purely translational motion (or removal of rotation from the motion field); and,

2. the model cannot be applied to approach angles close to fronto-parallel with the surface. In this case, the translational motion of the ground plane vanishes, and is replaced by second-order, diverging image motion.

It is possible to have alternate strategies on-board to deal separately with grazing and frontal approaches. However, it is preferable from both an engineering and control design stand point to have a single docking and landing control scheme applicable to any angle of approach, without modification.

4.7.4 Summary of time-to-contact for visuo-motor control

We have now reviewed techniques for estimating and applying time-to-contact, and discussed current issues/limitations imposed on its use for visuo-motor control. Tables 4.1 and 4.2 provide a summary of time-to-contact estimation techniques discussed in this chapter. Table 4.1 summarises techniques for estimating time-to-contact. Table 4.2

Table 4.2: Summary of time-to-contact schemes in the control loop

Restrictions/Assumptions	References
Fronto-parallel surface alignment	Tistarelli and Sandini [1993], Santos-Victor and Sandini [1997], Questa and Sandini [1996]
Motion and optical axis alignment	Ancona and Poggio [1993], Zufferey and Floreano [2006], Green *et al.* [2004], Coombs *et al.* [1998], Bermùdez *et al.* [2007], Questa and Sandini [1996]
Assumed/recovered egomotion and/or surface orientation	Nelson and Aloimonos [1989], Questa *et al.* [1995], Santos-Victor and Sandini [1996][1997], Lourakis and Orphanoudakis [1999], Cipolla and Blake [1997], Colombo [2000], Chahl *et al.* [2004], Ruffier and Francheschini [2005]
Near fronto-parallel and/or known surface orientation	**Chapter 5**: Robust to motion-optical axis misalignment, and surface orientation variation.
Planar surface	**Chapter 6**: Unified docking/landing with planar surfaces of arbitrary orientation.

groups work that has applied time-to-contact in the control loop according to the conditions in which it can operate (as imposed by the time-to-contact estimation scheme). This thesis proposes both new methods for estimating time-to-contact, and presents new control schemes for applying time-to-contact in the control loop. References to these chapters are included in the above tables, showing how this work addresses the limitations of previous work.

4.8 Summary

We have now motivated and reviewed background theory and literature on the visual estimation of contact from optical flow. In Chapter 2 we explored the role of vision in navigation, focussing specifically on robot navigation. From this, we prescribed a role for vision without global reconstruction and motivated an ecological approach to visual navigation using optical flow. In Chapter 3 we reviewed techniques for inferring scene structure and self-motion from the optical flow field, providing contact estimation under a traditional structure-from-motion framework. We discussed issues hindering the application of structure-from-motion techniques in real-time navigation systems, and reviewed arguments in favour of a biologically-inspired spherical projection model over a global field of view for robust structure-from-motion recovery.

In this chapter we have considered visual contact estimation for direct perception and visuo-motor control. We have motivated the use of time-to-contact as an important visual cue for visuo-motor control, with significant biological support. We have reviewed the theory, estimation and application of time-to-contact for visuo-motor control, and discussed limitations imposed on its use for robot navigation. We have argued that existing techniques for estimating time-to-contact do not adequately account for such limitations, thus restricting their application to tasks requiring minimal accuracy to meet the needs of control. Through this, we have highlighted a need for improved techniques for estimating time-to-contact with a specific focus on the needs of fine motion control under real world conditions, and in close proximity with surfaces. Chapters 5 and 6 of this thesis seek to improve on existing techniques for estimating and applying time-to-contact to visuo-motor control.

The following chapters present the contributions of this thesis. The first of these contributions is a proposed strategy for robustly estimating time-to-contact in the presence of noisy on-board conditions. This forms the basis of a divergence-based visuo-motor docking scheme for a ground-based mobile robot.

Chapter 5

Robust Visual Docking using Flow Field Divergence

5.1 Introduction

In Chapter 4 we showed how time-to-contact may be directly measured from the apparent expansion, or divergence, of optical flow vectors induced by motion towards an object (or an objects motion towards the observer). It thus provides a directly available alternative to explicit structure-from-motion solutions for visually guiding navigation with looming surfaces. While commonly applied to obstacle avoidance tasks, few have applied flow-based time-to-contact to tasks requiring fine motion control such as docking. Thus, a key issue for docking is achieving sufficiently robust time-to-contact estimates, capable of handling noisy on-board conditions throughout the manoeuvre.

In this chapter we present a novel, and robust strategy for docking a mobile robot in close proximity with an upright planar surface using optical flow field divergence. Unlike previous approaches, we achieve this without the need for explicit segmentation of the surface in the image, and using complete gradient-based optical flow estimation in the control loop (*i.e.,* no affine models, or flow field approximations are used to estimate the optical flow field). Central to the robustness of our approach is the derivation of a time-to-contact estimator that accounts for small rotations of the robot during ego-motion. This is achieved through tracking of the *focus of expansion* (FOE). We provide a theoretical justification for the constant tracking of the FOE as a means of accounting for not just the physical misalignment of the optical and translational axes, but also frame-to-frame shifts of the optical axis due to instantaneous rotations

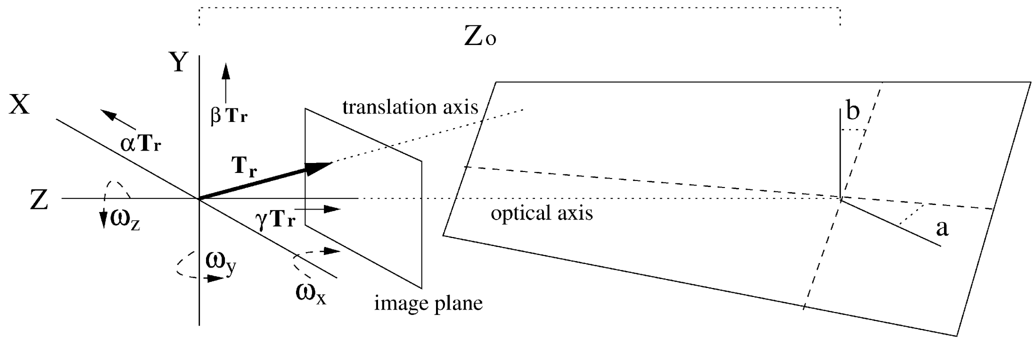

Figure 5.1: Geometric configuration

during ego-motion.

The chapter is structured as follows. Section 5.2 provides theoretical background, and the derivation of the proposed FOE-based time-to-contact estimator outlined above. We derive this for the specific case of controlling forward velocity towards a planar surface. In Section 5.3 we present a series of experiments providing quantitative assessment of the proposed time-to-contact estimation scheme in simulation, and over real image sequences. In addition, we examine performance in the control loop of a mobile robot docking with an upright planar surface. We follow this with a discussion of the results. Section 5.5 summarises the chapter.

5.2 Theory

5.2.1 Derivation of proposed time-to-contact estimator

The analysis presented here extends on the geometric modelling used by Santos-Victor and Sandini [1997]. We represent the docking surface as a plane in a camera centred coordinate system:

$$Z(X,Y) = Z_0 + aX + bY, \tag{5.1}$$

where Z_0 is the distance to the surface along the optical axis, X and Y represent points on the surface, and a and b give the slant and tilt with respect to the optical axis. By introducing the perspective projection equations into Equation 5.1, the surface plane can be expressed as a function of the image coordinates, (x, y)

[Santos-Victor and Sandini 1994]:

$$Z(x,y) = \frac{Z_0}{1 - a\frac{x}{f_\mathrm{x}} - b\frac{y}{f_\mathrm{y}}}, \tag{5.2}$$

where f_x and f_y are focal lengths expressed in pixels.

Given a fixed camera with respect to the robot's direction of motion, we represent the translational velocity of the camera, T_c, as proportions of the forward translational velocity, T_r, of the robot:

$$T_c = \begin{bmatrix} \alpha T_r & \beta T_r & \gamma T_r \end{bmatrix}. \tag{5.3}$$

The camera's angular velocity (ω_c) is given by:

$$\omega_c = \begin{bmatrix} \omega_x & \omega_y & \omega_z \end{bmatrix}, \tag{5.4}$$

where each component represents rotation about the axis indicated by its subscript. Figure 5.1 shows the geometric configuration.

The optical flow induced by the apparent motion of the docking plane is defined by the well known equations [Santos-Victor and Sandini 1997]:

$$\begin{aligned} u(x,y) &= f_\mathrm{x}\left[\frac{\gamma T_r(\frac{x}{f_\mathrm{x}} - \alpha)}{Z(x,y)} + \omega_x \frac{xy}{f_\mathrm{x} f_\mathrm{y}} - \omega_y(1 + \frac{x^2}{f_\mathrm{x}^2}) + \omega_z \frac{y}{f_\mathrm{y}}\right], && (5.5)\\ v(x,y) &= f_\mathrm{y}\left[\frac{\gamma T_r(\frac{y}{f_\mathrm{y}} - \beta)}{Z(x,y)} + \omega_x(1 + \frac{y^2}{f_\mathrm{y}^2}) - \omega_y \frac{xy}{f_\mathrm{x} f_\mathrm{y}} - \omega_z \frac{x}{f_\mathrm{x}}\right], && (5.6) \end{aligned}$$

where $u(x,y)$ and $v(x,y)$ are the horizontal and vertical components of motion.

Let us now consider the effects of rotation, causing the FOE to shift with respect to the optical axis. Let (x', y') be an arbitrary point in the image representing the FOE. We define the depth of the surface, $Z(x,y)$, with respect to the FOE:

$$Z(x,y) = \frac{Z(x',y')}{1 - a\frac{(x-x')}{f_\mathrm{x}} - b\frac{(y-y')}{f_\mathrm{y}}}. \tag{5.7}$$

Substituting (5.7) into Equations 5.5 and (5.6), we obtain:

$$u(x,y) = \frac{\gamma T_r(x - f_\mathrm{x}\alpha)}{Z(x',y')}\left[1 - \frac{a(x-x')}{f_\mathrm{x}} - \frac{b(y-y')}{f_\mathrm{y}}\right] + \omega_x\frac{xy}{f_\mathrm{y}} - \omega_y(f_\mathrm{x} + \frac{x^2}{f_\mathrm{x}}) + \omega_z\frac{y}{f_\mathrm{x}}, \tag{5.8}$$

$$v(x,y) = \frac{\gamma T_r(y - f_\mathrm{y}\beta)}{Z(x',y')}\left[1 - \frac{a(x-x')}{f_\mathrm{x}} - \frac{b(y-y')}{f_\mathrm{y}}\right] + \omega_x(f_\mathrm{y} + \frac{y^2}{f_\mathrm{y}}) - \omega_y\frac{xy}{f_\mathrm{x}} - \omega_z\frac{x}{f_\mathrm{x}}. \tag{5.9}$$

Given the optical flow at the FOE is zero, substituting for $x = x'$ and $y = y'$ provides the following constraints on the optical flow at the FOE:

$$0 = \frac{\gamma T_r(x' - f_\mathrm{x}\alpha)}{Z(x',y')} + \omega_x\frac{x'y'}{f_\mathrm{y}} - \omega_y(f_\mathrm{x} + \frac{x'^2}{f_\mathrm{x}}) + \omega_z\frac{y'}{f_\mathrm{x}}, \tag{5.10}$$

$$0 = \frac{\gamma T_r(y' - f_\mathrm{y}\beta)}{Z(x',y')} + \omega_x(f_\mathrm{y} + \frac{y'^2}{f_\mathrm{y}}) - \omega_y\frac{x'y'}{f_\mathrm{x}} - \omega_z\frac{x'}{f_\mathrm{x}}. \tag{5.11}$$

Solving for ω_x and ω_y, we obtain:

$$\omega_x = \frac{f_\mathrm{y}}{x'y'}\left[\frac{\gamma T_r}{Z(x',y')}(x' - f_\mathrm{x}\alpha) + \omega_y(f_\mathrm{x} + \frac{x'^2}{f_\mathrm{x}}) + \omega_z\frac{y'}{f_\mathrm{y}}\right], \tag{5.12}$$

$$\begin{aligned}\omega_y = &\frac{1}{f_\mathrm{x}f_\mathrm{y}(1 + \frac{x'^2}{f_\mathrm{x}^2} + \frac{y'^2}{f_\mathrm{y}^2})}\left[\frac{T_r}{Z(x',' y')}\Big(x'y'\beta + f_\mathrm{y}x' + \right.\\ &\left. f_\mathrm{x}f_\mathrm{y}\alpha + \frac{f_\mathrm{x}\alpha y'^2}{f_\mathrm{y}}\Big) - \omega_z\Big(y' + \frac{y'^3}{f_\mathrm{y}^2} - \frac{x^2y'}{f_\mathrm{x}f_\mathrm{y}}\Big)\right].\end{aligned} \tag{5.13}$$

Taking the partial derivatives of Equations 5.8 and (5.9) in their respective directions, and again substituting for $x = x'$, $y = y'$, we obtain the partial derivatives at the FOE, defined as:

$$\left.\frac{\partial u}{\partial x}\right|_{\mathrm{foe}} = \frac{\gamma T_r}{Z(x',y')}\left[1 - a\left(\frac{x'}{f_{\mathrm{x}}} + \alpha\right)\right] + \omega_x \frac{y'}{f_{\mathrm{y}}} - \omega_y \frac{2x'}{f_{\mathrm{x}}}, \tag{5.14}$$

$$\left.\frac{\partial v}{\partial y}\right|_{\mathrm{foe}} = \frac{\gamma T_r}{Z(x',y')}\left[1 - b\left(\frac{y'}{f_{\mathrm{y}}} + \beta\right)\right] + \omega_x \frac{2y'}{f_{\mathrm{y}}} - \omega_y \frac{x'}{f_{\mathrm{x}}}. \tag{5.15}$$

Summing these, we obtain the flow field divergence at the FOE (D_{foe}):

$$D_{\mathrm{foe}} = \frac{-\gamma T_r}{Z(x',y')}\left[a\left(\frac{x'}{f_{\mathrm{x}}} + \alpha\right) + b\left(\frac{y'}{f_{\mathrm{y}}} + \beta\right) - 2\right] + 3\left(\frac{\omega_x y'}{f_{\mathrm{y}}} - \frac{\omega_y x'}{f_{\mathrm{x}}}\right), \tag{5.16}$$

and from this we obtain an equation for the relative depth of the scene point projecting to the FOE:

$$\frac{Z(x',y')}{T_r} = \frac{\gamma}{D_{\mathrm{foe}}}\left[a\left(\frac{x'}{f_{\mathrm{x}}} + \alpha\right) + b\left(\frac{y'}{f_{\mathrm{y}}} + \beta\right) - 2\right] - \frac{3Z(x',y')}{D_{\mathrm{foe}} T_r}\left(\frac{\omega_x y'}{f_{\mathrm{y}}} - \frac{\omega_y x'}{f_{\mathrm{x}}}\right) \tag{5.17}$$

Using Equations 5.12 and 5.13, we substitute for ω_x and ω_y in (5.17) and thus remove both rotations from (5.17) such that:

$$\begin{aligned}\frac{Z(x',y')}{T_r} = \frac{\gamma}{D_{\mathrm{foe}}}\Bigg[&1 + a\left(\frac{x'}{f_{\mathrm{x}}} + \alpha\right) + b\left(\frac{y'}{f_{\mathrm{y}}} + \beta\right) \\ &- \frac{3}{\gamma x'}\Bigg(- f_{\mathrm{x}}\alpha + \frac{(x' f_{\mathrm{y}} + f_{\mathrm{x}} f_{\mathrm{y}}\alpha + x'y'\beta + \frac{y'^2 f_{\mathrm{x}}\alpha}{f_{\mathrm{y}}})}{f_y(1 + \frac{x'^2}{f_{\mathrm{x}}^2} + \frac{y'^2}{f_{\mathrm{y}}^2})} \\ &+ \frac{\omega_z y' T_r}{f_{\mathrm{y}} Z(x',y')}\Big(f_{\mathrm{y}} + \frac{y'^2}{f_{\mathrm{y}}} - \frac{x'^2}{f_{\mathrm{x}}} - 1\Big)\Bigg)\Bigg].\end{aligned} \tag{5.18}$$

Notably, the removal of ω_x and ω_y introduces a term involving camera roll (ω_z). If required, techniques for roll removal such as that of Hanada and Enjima [2000] can also be applied without prior knowledge of the rotation.

Equation 5.18 gives the relative depth of the scene point projecting to the FOE. If T_r is aligned with the FOE, then (5.18) also gives a precise measure of time-to-contact. In the presence of rotations, however, this assumption is unlikely to hold.

However, considering a docking scenario for a finite sized robot, the presence of small instantaneous rotations will also mean that the precise point of impact is unlikely to be known. Given that the FOE provides the only location in the flow field where rotation is accounted for, we can consider (5.18) to be a reasonable approximation of time-to-contact (referred to as τ_{foe}) under these conditions.

5.2.2 Time-to-contact for a ground-based mobile robot

Consider Equation 5.18 for the case of a mobile robot, moving on a ground plane towards a visible planar surface. Given a fixed, approximately forward facing camera, ω_z will be negligible and can therefore be set to zero. In addition, the camera orientation parameters with respect to the heading direction: α, β and γ, can also be set to known values ($\alpha = \beta = 0, \gamma = 1$), thus reducing Equation 5.18 to:

$$\tau_{foe} = \frac{1}{D_{\mathrm{foe}}} \left[1 + \frac{ax'}{f_{\mathrm{x}}} + \frac{by'}{f_{\mathrm{y}}} - \frac{3}{(\frac{x'^2}{f_{\mathrm{x}}^2} + \frac{y'^2}{f_{\mathrm{y}^2}} + 1)} \right]. \tag{5.19}$$

Note that the only potential unknowns in Equation 5.19 are the surface orientation parameters: a and b. If unknown, these parameters form a bound on τ_{foe}, such that:

$$\tau_{\mathrm{foe}} = \frac{1}{D_{\mathrm{foe}}} \left[1 - \frac{3}{(\frac{x'^2}{f_x} + \frac{y'^2}{f_y} + 1)} \right] \pm \frac{1}{D_{\mathrm{foe}}} \left[\frac{ax'}{f_x} + \frac{by'}{f_y} \right]. \tag{5.20}$$

Note that the surface orientation terms represent the deformation components of the time-to-contact.

5.2.2.1 Constraints on rotation

We now consider the effect of rotation on both the existence of the FOE, and its location in the image. In particular, we seek to define the range of allowable shifts of the FOE from the image centre, given a known amplitude of expected rotations during egomotion.

We first consider the location of the FOE. For a ground-based robot with forward facing camera, and rotation only about the Y axis (ω_y), we consider only horizontal shifts of the FOE. We re-examine the equation for the horizontal component of optical flow in Equation 5.10. Solving for x', we obtain the following solutions:

$$x' = \frac{f_x}{2}\left[\frac{T_r}{Z(x')\omega_{\mathrm{y}}} \pm \sqrt{\left(\frac{-T_r}{Z(x')\omega_y}\right)^2 - 4}\right]. \tag{5.21}$$

Notably, two solutions exist, representing the two locations where flow is zero in the infinite image plane: at the FOE and at the FOC (focus of contraction). However, given a forward facing camera, the FOE will always be the minimum of the two solutions (*i.e.,* closest to the image centre).

The square root term in Equation 5.21 defines a constraint on the maximum allowable shift of the FOE from the image centre. Specifically:

$$\left|\frac{T_r}{Z(x')\omega_y}\right| \geq 2. \tag{5.22}$$

The term $\frac{T_r}{Z(x')\omega_y}$ defines the ratio of scaled velocity and rotation. It can be seen that as rotation grows large with respect to translation, the ratio decreases. The limit of this decrease is given by $\frac{T_r}{Z(x')\omega_y} = 2$, after which, no real solution for the location of the FOE exists. Substituting $\frac{T_r}{Z(x')\omega_y} = 2$ into Equation 5.21, we obtain a unique solution for the FOE location:

$$x' = fx. \tag{5.23}$$

Thus, the maximum allowable displacement of the FOE from the image centre is the same as the focal length (*i.e.,* $\pm 45^o$ from the image centre). Naturally, if the horizontal field of view is narrower than 90^o, or physical limitations require it, this range may be further restricted. Note that the unique solution for x' obtained from this substitution represents the convergence of the FOE and FOC to a single location in the image. As rotation grows larger, the two points approach each other. Thus, the FOE shift limit is equivalently defined by this convergence.

From Equation 5.22, we also note the following constraint on τ:

$$\tau^{-1} = \frac{T_r}{Z(x')} \geq 2\omega_y \tag{5.24}$$

The above highlights an important secondary purpose for the velocity control design. While maintaining a safe speed of approach is the primary goal, $\tau^{-1} = \frac{T_r}{Z(x')}$ must also be kept sufficiently high to ensure FOE shifts remain within the required bounds. Thus, for a given amplitude of expected rotations during egomotion, a reference τ^{-1} must be set to ensure that such rotations fall within these bounds.

5.2.2.2 Constraints on angle of approach (surface orientation)

If the angle of approach is roughly fronto-parallel, then a and b will be small. Thus, the bound on τ_{foe} should be well contained. As a and b increases, the bound on τ_{foe} also increases. Time-to-contact estimates are therefore unlikely to remain stable for eccentric approach angles. Even if the angle of approach is roughly known, errors between the assumed approach angle and actual approach angle will be exacerbated if the approach angle is significantly away from fronto-parallel. Thus, while the proposed time-to-contact scheme provides tolerance to deviations from the intended approach, it is best suited to near fronto-parallel approach angles.

In order to achieve docking with an object surface, the FOE must lie within the imaged region of the target surface area. Therefore, ensuring the FOE always exists within the projected surface target area should maintain an approach angle that is within stability limits. This may also be used as a means of assessing the achievability of the task.

5.3 Experimental Results

In this section we present four sets of experiments demonstrating the performance of the proposed FOE-based time-to-contact estimator to the task of docking. We provide results from simulation, off-board image sequences, and from the technique's application to the closed-loop control of a mobile robot performing a docking manoeuvre. We first describe each experiment and discuss issues relating to the application of the FOE-based time-to-contact strategy. We then present the results of these experiments.

5.3.1 Simulation experiment

A simulation was conducted modelling the motion of a ground-based mobile robot, with camera, towards a planar fronto-parallel surface. A 2D motion model was used, allowing only forward velocity and a single rotation in the ground plane. As such, only the u component of flow across a single row of pixels was required to obtain time-to-contact estimates. From this, a set of sample flow fields were obtained.

For each consecutive sample, the distance to the surface was decremented by a constant amount. The robot was assumed to be initially aligned fronto-parallel with the surface before a constant translational velocity, and randomly selected instantaneous rotational velocity were applied to the scene with respect to the robot's location. The resulting motion vectors were then projected onto the robot's image plane, thereby generating the expected flow resulting from the robot's motion with respect to the scene. From this, the FOE (which shifts as a result of the rotation) was located, and time-to-contact computed using (5.19). The simulation was implemented and run in a Matlab environment.

5.3.2 Off-board time-to-contact experiments

5.3.2.1 Indoor image sequence

A real image *looming wall sequence* depicting the motion of a camera towards a textured planar surface was constructed. Figure 5.2(a) shows sample frames from the sequence. In the construction of the image sequence, the camera was moved 3cm per frame towards a heavily textured, approximately fronto-parallel wall. Optical flow fields were estimated for each frame of the sequence, and from this, time-to-contact estimates obtained.

Flow divergence was estimated using four patches in the image, each centred on a distance of 12 pixels from the FOE, and each at 45 degrees from the horizontal and vertical axes that intersect at the FOE. Figure 5.2(a) shows this patch configuration. For comparison, time-to-contact was also estimated by placing the four patches about the image centre.

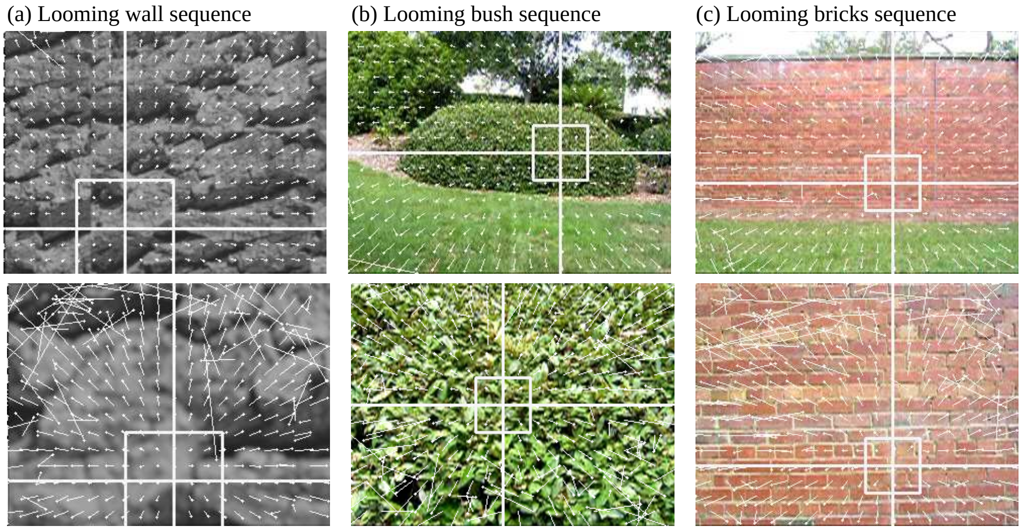
(a) Looming wall sequence
(b) Looming bush sequence
(c) Looming bricks sequence

Figure 5.2: Sample frames and flow fields from each image sequence used in off-board experiments: (a) *looming wall*, (b) *looming bush*, and (c) *looming bricks*. Line intersections show estimated FOE for frame, and boxes indicate the divergence patch configurations used for FOE-based time-to-contact estimation.

5.3.2.2 Outdoor image sequences

To test the technique's robustness under more natural conditions, two outdoor image sequences were constructed, depicting the motion of the camera towards different, more natural surfaces. In both sequences, the camera was mounted on the handle bars of a bicycle, equipped with a speedometer to gauge the approximate speed of approach. The bike was rolled at constant velocity towards both surfaces. Figures 5.2(b) and (c) show sample frames from both sequences: the *looming bush sequence*, and the *looming bricks sequence*. Note that the camera's motion was subject to rotations induced by the uneven terrain (including camera roll), and small adjustments of heading. The initial distance in both sequences was 9m. The average velocity of the camera depicted in the *looming bush sequence* is approximately 13cm per frame (6km/hr), and 20.5cm per frame (8.5km/hr) for the *looming bricks sequence*.

Flow divergence was estimated from optical flow vectors within a single 51×51 pixel patch centred on the estimated location of the FOE. Divergence was also measured at the image centre using the same patch size.

5.3.3 On-board docking experiment

To test the robustness of the FOE-based time-to-contact measure, the technique was integrated into a simple closed-loop docking behaviour for velocity control of a mobile robot. In the experiment, a robot with a single, fixed, forward facing camera approached a heavily textured, roughly fronto-parallel wall, attempting to decelerate and safely stop as close to the wall as possible without collision. Figure 5.3 shows the experimental workspace.

The robot used is velocity controlled, that is, the control signal is passed to a servo motor that controls the rolling speed of the drive wheels. Initial experimental tests showed that direct proportional feedback of the drive wheels lead to highly aggressive control action due to the noise in the divergence measure. By incorporating a virtual model of robot dynamics in the control design, the closed-loop behaviour of the vehicle was smooth and well conditioned. The discrete time realisation of the proposed control law is

$$v_t = \Delta v_{t-1} + \frac{\Delta K_p}{m}(D_{\text{ref}} - D_t), \tag{5.25}$$

where $v(t)$ is the velocity control input at time t, Δ is the discretisation time, m is a virtual vehicle mass, K_p is a proportional gain, D_t is the most recent flow divergence estimate, and D_{ref} is the reference set-point for flow divergence ($\frac{\Delta K_p}{m} = 0.0325$ and $D_{\text{ref}} = 0.022$ for these trials). Along with the discrete-time kinematics

$$z_t = \Delta v_{t-1}. \tag{5.26}$$

Flow divergence was estimated using two 40×40 pixel image patches, each placed at 45 degrees on either side of the vertical axis passing through the FOE, and each centred on a distance of 25 pixels from the FOE. The patches were placed only above the FOE to avoid measuring divergence on the imaged ground plane. Reasons for the variation of patch size and configuration used in the off-board experiment were based on empirical observations of performance on-board. Due to the noisier conditions on-board, larger patch sizes were used to obtain a more robust estimate of flow divergence during ego-motion. In general, a range of patch sizes and configurations were found to obtain strong results.

5.3.4 Optical flow and FOE estimation

Throughout all experiments over real image sequences (including on-board), Lucas and Kanade's [1981] gradient-based method was applied. For the indoor *looming wall* sequence, a standard implementation of Lucas and Kanade's algorithm was applied, and flow vectors were obtained for all image points. Due to significantly larger flow experienced in both outdoor sequences, the pyramidal implementation of Lucas and Kanade's technique (outlined in Section 3.2.2.5) was applied. To offset the increased computation load of this approach, flow vectors were only estimated for every fifth pixel.

In all experiments, the FOE was calculated using a simple algorithm that requires the imaged surface to occupy the entire viewing field (or at least, the section of the viewing field for which the FOE is expected to lie within). To obtain x', each row in the image was used to count the number of positive and negative horizontal flow components, which were then differenced, and averaged over all rows to locate the overall zero point for x. The algorithm was applied similarly to obtain y', using the signs of vertical components of flow. While more sophisticated algorithms for locating the FOE do exist, it is important to note that in many cases, pure (or close to pure) translational motion is assumed (*e.g.,* [Sazbon et al. 2004] [Negahdaripour. 1996] [Jain 1983], see Section 3.4.2). In contrast, the technique applied here provides a relatively high tolerance to rotation, such that the FOE will always be located so long as it lies within the imaged area, and other local minima in the flow field do not exist. Given only the sign of flow vectors are used to estimate the FOE, the computation associated with its estimation is negligible in comparison with the flow estimation itself. It should be noted that other suitable techniques do exist, such as [Li 1992], that do not require the segmentation of the object surface area. The algorithm employed here was chosen primarily for its efficiency in achieving reasonably accurate FOE estimates.

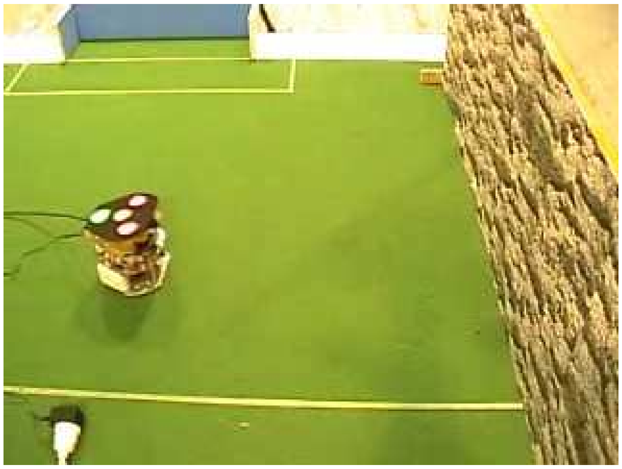

Figure 5.3: Setup for on-board docking tests.

5.3.5 Results

5.3.5.1 Simulation results

Figure 5.4 gives the simulation results, showing a direct comparison of time-to-contact obtained using the FOE-based estimator defined by Equation 5.19, and estimates obtained from the measured divergence at the image centre (using $\frac{2}{\text{div}}$). Ground truth time-to-contact is also provided, computed from the robot's distance from the surface and its known constant forward velocity towards the surface. It can be seen that the FOE-based time-to-contact measure closely reflects ground truth. Small discrepancies between the FOE-based measure and ground truth are the result of unavoidable quantisation errors in the image, disallowing the precise location of the FOE.

In contrast, time-to-contact estimates taken along the optical axis exhibit significant fluctuation compared with that obtained at the FOE. It is also evident that the image centre always provides an over estimate of time-to-contact, a result of the optical axis deviating from its fronto-parallel alignment with the surface. While errors in time-to-contact are reduced as the distance to the surface approaches zero, it is important to note that this is due to the robot's constant velocity towards the surface. As the surface draws near, the translational flow increases, thereby diminishing the effects of the robot's rotation in the flow field.

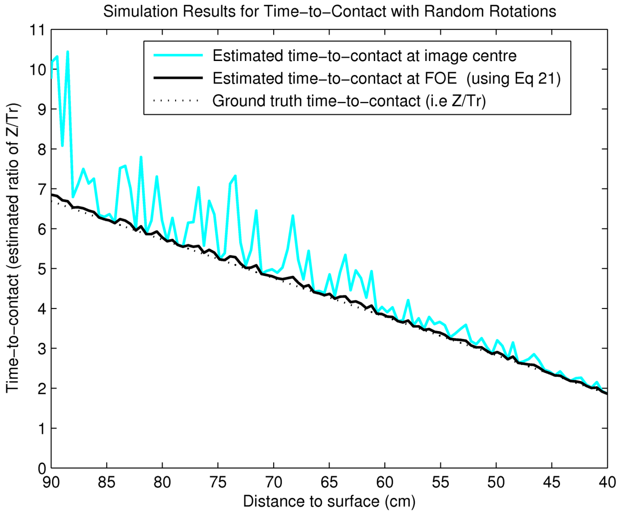

Figure 5.4: Simulation results compare our FOE-based time-to-contact estimator (Equation 5.19), with time-to-contact estimates obtained at the image centre using $\frac{2}{\text{div}}$ for the simulated 2D motion of a ground-based mobile robot translating at constant speed towards a fronto-parallel, planar surface. For each sample, the robot's forward speed, and randomly chosen instantaneous rotational velocity ($-0.1 \leq \omega_y \leq 0.1$) were used to compute the corresponding horizontal flow. From this, time-to-contact estimates were obtained. Ground truth shows the exact time-to-contact for each sample, given the robot's forward velocity and distance from the surface. For all samples, the camera's focal length is set to 188px.

5.3.5.2 Indoor image sequence results

Figure 5.5(a) shows time-to-contact estimates for each frame of the indoor looming-wall sequence for the FOE-based, and image-centre-based strategies. Ground truth time-to-contact is also shown, obtained from the camera's known velocity, and a best linear fit over time-to-contact measures obtained from ground truth flow fields constructed from camera calibration.

From these results, a significant improvement in the consistency of time-to-contact estimates is achieved when divergence is calculated with respect to the FOE. Of particular note, the FOE-based strategy achieves a close match with ground truth from the fifteenth frame onward. In contrast, the image centre-based method consistently over-estimates time-to-contact, and exhibits larger fluctuations across the sequence.

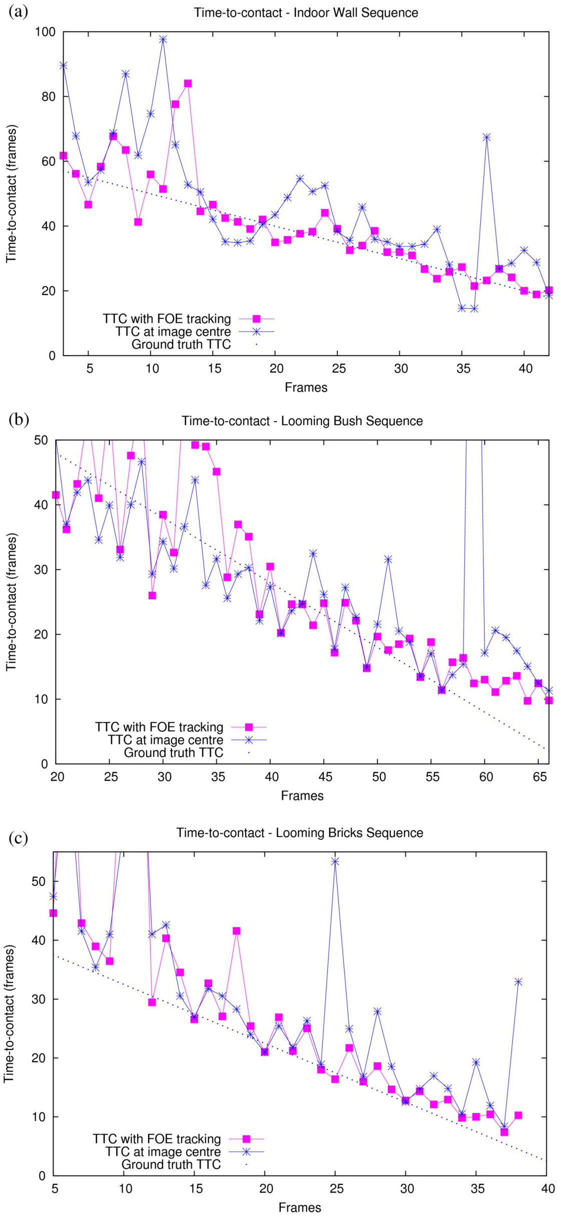

Figure 5.5: Time-to-contact estimates for: (a) indoor looming wall sequence, (b) looming bush sequence, and (c), looming bricks sequence.

5.3.5.3 Outdoor image sequence results

Figures 5.5(b) and (c) show time-to-contact estimates for both outdoor image sequences, again comparing the FOE-based, and image-centre-based strategies.

As with the indoor *looming wall sequence*, improvements in time-to-contact estimation are achieved by the FOE-based strategy as the surface approaches. This is evident from frame 40 onward for the *looming bush* sequence, and from frame 20 in the *looming bricks* sequence.

Across all sequences, larger fluctuations are evident in early frames for both strategies. This is unsurprising given the flow due to camera translation is unlikely to be large enough to be reliably measured at this distance from the wall. It is also likely that the FOE is poorly defined at this distance. In early frames of both outdoor sequences, the FOE's location was observed to shift significantly, and in some cases (particularly for the *looming bush* sequence), fall outside the imaged area of the surface. As divergence increases, however, the FOE-based strategy quickly stabilises, and begins to outperform the image-centre-based estimator.

5.3.5.4 On-board docking results

Six trials of the FOE-based docking strategy were conducted, and data recorded. Video 5.1 on the thesis CD-ROM provides footage of a sample trial from on-board trials. Figure 5.6 shows the velocity-distance profiles and the plotted approach of the robot towards the surface for each trial. Also shown is the theoretically expected velocity-distance profile based on the integrating of Equation 5.25 in discrete time for the initial velocity, distance and tuning parameter values used in the trials. Of the six trials conducted, the FOE-based strategy docked in close proximity to the surface five times without collision. Only one collision, Trial 2, was observed. Results shown in Figure 5.6 suggest this was most likely due to noise effected divergence estimates obtained around 30cm from the surface.

Notably, results show an early lack of response compared with the predicted deceleration. This is a likely result of divergence being too small to measure at such distances. As the robot approaches, the measured divergence increases, and the velocity-distance

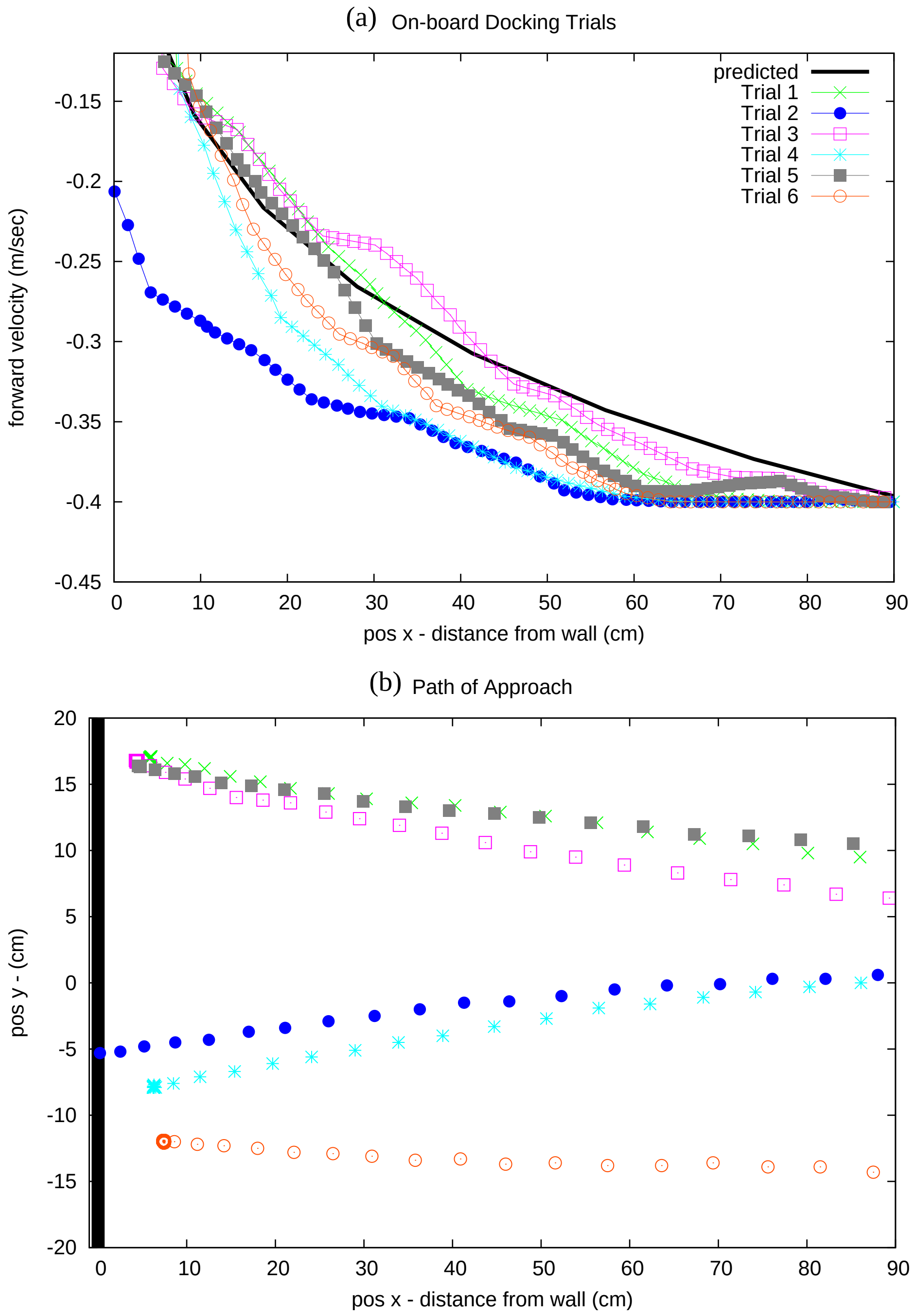

Figure 5.6: On-board docking results showing (a) velocity-distance profiles, and (b), the plotted paths of the robot for each trial.

profiles begin to resemble theoretical expectations.

Figure 5.6(b) shows considerable variation in both the robot's initial starting position, and the extent (and direction) of the lateral drift experienced during each approach. Despite the presence of rotational motion and varying angles of approach, highly robust, close-proximity stopping distances were still achieved. The average stopping distance achieved over the successful trials was 6cm, with the furthest distance recorded being just 7cm.

An attempt was made to compare the FOE-based on-board control scheme with the same control scheme using an image-centre based divergence measure. The raw divergence estimates obtained at the image-centre, however, were found to be unworkable for the simple proportional control scheme used. A large range of tuning parameter values were explored.

5.4 Discussion

Among the successful trials, close proximity stopping distances were achieved with higher than expected consistency. Recorded velocity-distance profiles, and stopping distances are also consistent with theoretical expectation. The consistency achieved in stopping distances is encouraging when considering the simple control law used, and significant differences in the plotted approach path of the robot during each trial. Notably, however, one fail case (trial 2) was recorded. More sophisticated control schemes, and higher update frequencies would be expected to further improve the robustness of this strategy.

The FOE-based docking strategy compares well with previous work in flow-based docking. The final stopping distances achieved are a significant improvement on Questa *et al.* [1995] (approximately 15cm), and comparable with Santos-Victor and Sandini [1997]. Unlike previous work, we report highly consistent results over a set of trials. In addition, we obtain these results using general optical flow estimation (no affine approximations), and without filtering of the divergence estimates. We acknowledge, however, that we are using newer and faster computers than in previous work, thus allowing faster estimation of the optical flow.

An acknowledged drawback of the proposed docking scheme is the necessity of the FOE to exist within the image plane. To maintain this condition, it was necessary to ensure sufficient translational motion existed to counter potential shifts of the FOE beyond image boundaries (as per Equation 5.24). Across all image sequences, the FOE was observed to shift significantly, despite the predominantly translational motion. It is important to note, however, that constraints on rotation are a function of camera focal length, as well as the ratio of translation and rotation. Thus, tolerance to rotations is likely to be increased with wider angle cameras.

In addition to handling rotational effects, the FOE-based strategy was observed to provide increased robustness to flow exceeding measurable levels in each sequence. This effect is evident in flow fields shown in the bottom row of Figure 5.2, where peripheral flow vectors become noisy and unreliable. While generally only in the periphery, this region of flow becomes larger as $\frac{Tr}{Z}$ increases (*i.e.,* $Z \rightarrow 0$). As a result, any shifting of the FOE when in close proximity to the surface may cause this region of large flow to inhabit image-centre-based divergence patches. This is the likely cause of larger fluctuations in image-centre-based time-to-contact estimates in the later frames of each sequence (particularly for the *looming bricks* sequence, where forward velocity was significantly faster). In contrast, time-to-contact estimates taken with respect to the FOE remain stable under these conditions, and in accordance with simulation results, appear to improve in consistency as Z decreases. This improvement appears also to result from the FOE itself being more clearly defined, and therefore more accurately located when $\frac{Tr}{Z}$ is large.

A key advantage of the proposed time-to-contact estimator is its improved tolerance to surface misalignment due to rotation. While in theory, the estimator may be applied for any known angle of approach, the results do not support its use as a general docking solution for any surface orientation. As discussed in Section 5.2.2.2, stability degrades rapidly as the angle of approach moves away from fronto-parallel. Most prominent is the effect of unintentional rotations, causing greater errors as the approach angle shallows. While increasing the translational motion component, or the field of view should improve tolerance levels, approaches will always be less stable away from a fronto-parallel approach.

5.5 Summary

This chapter has presented a mobile robot docking strategy that utilises a time-to-contact estimation that is robust to noisy, instantaneous rotations induced by robot ego-motion. We have shown that through tracking the focus of expansion (FOE) in the optical flow field, small rotations of the camera and misalignments of the optical and translational axes can be accounted for by calculating flow divergence with respect to the FOE. In this way, the effects of the rotation are effectively cancelled out, and improved accuracy and stability is achieved. Based on this, we have proposed a divergence-based visuo-motor control scheme for docking a robot with near fronto-parallel surfaces, verified though experimental trials. These results show a significant improvement in the consistency and robustness of time-to-contact estimates when compared with common strategies that take no account of the shifting FOE during robot ego-motion. The FOE-based time-to-contact estimator was demonstrated to be sufficient for fine motion control of a mobile robot when in close proximity with the docking surface.

In the next chapter we explore the application of flow field divergence to docking with a surface of arbitrary orientation. In so doing, we propose a unified visuo-motor solution for docking and landing.

Chapter 6

Unifying Docking and Landing using Spherical Flow Divergence

6.1 Introduction

In Chapter 5 we proposed a visuo-motor control scheme for docking a mobile robot with an upright planar surface. We showed that computing divergence at the FOE accounts for small rotations during the approach. While alleviating strict assumptions of alignment with the surface, the algorithm is still constrained to near fronto-parallel approaches. These limitations are a direct consequence of the perspective projection model employed. Addressing the need for *non*-fronto-parallel docking/landing, we discussed in Chapter 4 (Section 4.7.3) a proposed insect-inspired model for performing graze landings from optical flow [Srinivasan et al. 2000]. In this model, the speed of approach is reduced by holding the angular motion of the ground plane constant, allowing a safe touch-down to be achieved. However, this model cannot be applied to fronto-parallel approaches, and assumes translational motion only. Thus, neither approach constitutes a general solution to the docking/landing problem.

In this chapter we present a novel strategy for landing and docking with planar surfaces of *arbitrary orientation*. Central to this strategy is the use of a spherical projection model, and the location of the divergence maximum on the view sphere (referred to as the *max-div* point). We show that limitations imposed under a perspective projection are not present under a spherical projection model, allowing robust estimates of τ for any angle of approach. Thus, for spherical cameras, the proposed control scheme supersedes the FOE-based scheme presented in Chapter 5. For a view sphere approaching a planar surface, we show that the max-div point will always be located half way along the arc connecting the direction of translation, and the planar surface normal. While Koenderink and Van Doorn [1976, 1981] first noted this property some time ago, subsequent interest has been constrained to the psychophysics literature and its possible role in human self-motion perception [1982]. To our knowledge, no one has considered the max-div point as a visual control input, or provided a formal proof of this property. The work presented therefore represents a novel application of the max-div point to velocity and heading control, and provides the first demonstration of significant advantages gained through its use for visuo-motor docking and landing.

The chapter is structured as follows. Section 6.2 provides a full derivation of time-to-contact on the view sphere, and a proof of the max-div property. Section 6.3 presents the proposed max-div visuo-motor control scheme for unifying docking and landing. Section 6.4 presents open-loop testing and experimental validation of the proposed time-to-contact estimation scheme over pre-constructed image sequences. Section 6.5 examines the closed-loop performance of the control scheme in performing controlled approaches to a surface of varying orientation. Experiments are conducted both in simulation and on-board a mobile robot. Section 6.6 provides an overall discussion of these results. Section 6.7 briefly presents a discussion of implications and broader outcomes for visuo-motor control schemes. Finally, Section 6.8 summarises the contributions of this chapter.

6.2 Theory

In this section we examine divergence (and time-to-contact) under a spherical projection model. We consider this in the context of a view sphere approaching a planar surface of arbitrary orientation. We discuss distinct advantages gained using a spherical projection model, and provide a formal proof of the max-div property.

6.2.1 Computing time-to-contact for planar surfaces

Recall from Chapter 3 (Section 3.3.2.3), the general equation for divergence on the unit view sphere [Koenderink and Doorn 1975]:

$$D\Big(\hat{p}\Big) = -\frac{v(\hat{p}\cdot\hat{t})}{R(\hat{p})}\left[1 + \frac{R'(\hat{p})}{R(\hat{p})}\Big(\frac{\hat{t}}{(\hat{p}\cdot\hat{t})} - \hat{p}\Big)\right], \tag{6.1}$$

where $\hat{p} \in \mathbb{R}^3$ is the direction of an arbitrary point on the sphere, $\hat{t}$ is the direction of motion of the sphere and v its velocity, and $R(\hat{p})$ is a radial depth function defining the distance to the point in space in the direction $\hat{p}$, and $R'(\hat{p})$ its derivative.

We now consider the case of a sphere approaching a single planar surface of arbitrary orientation. We define the radial distance to a point on the surface of a plane, R_{p}, projecting to a point $\hat{p} \in \mathbb{R}^3$ by the depth function:

$$R_{\hat{p}}(p) = \frac{R_o}{(\hat{p}\cdot\hat{n})}, \tag{6.2}$$

where $\hat{n} \in \mathbb{R}^3$ gives the direction of the planar surface normal on the sphere (*i.e.,* the closest surface point), and $R_o \in \mathbb{R}$ is the distance to this point. Substituting (6.2) into (6.1), we obtain:

$$D(\hat{p}) = -\frac{v}{R_o}\Bigg(2(\hat{p}\cdot\hat{t})(\hat{p}\cdot\hat{n}) - (\hat{n}\cdot\hat{t})\Bigg). \tag{6.3}$$

Through simple algebraic manipulation of (6.3), we obtain the following equation for time-to-contact when approaching a planar surface:

$$\tau_p(\hat{p}) = \frac{R_o}{v(\hat{p}\cdot\hat{t})(\hat{p}\cdot\hat{n})} = -\frac{1}{\mathrm{div}(\hat{p})}\Big(2 - \frac{(\hat{n}\cdot\hat{t})}{(\hat{p}\cdot\hat{t})(\hat{p}\cdot\hat{n})}\Big). \tag{6.4}$$

6.2.1.1 Advantages of a full view sphere for robust time-to-contact estimation

There exists a duality in the time-to-contact along the direction of translation, and the time-to-contact along the direction of the closest surface point. Substituting $\hat{p}$ for $\hat{t}$ or

$\hat{n}$ in Equation 6.4 yields:

$$\tau_p(\hat{t}) = \tau_p(\hat{n}) = \frac{1}{\mathrm{div}(\hat{t})} = \frac{1}{\mathrm{div}(\hat{n})}. \tag{6.5}$$

Thus, τ_p can be measured directly from the flow divergence in the direction of motion, $\hat{t}$, and the direction of the surface normal, $\hat{n}$. This highlights significant advantages over time-to-contact estimation under a perspective projection model. Table 6.1 sets out these advantages explicitly.

Spherical projection	Perspective projection
τ is precisely computable from divergence at *two distinct locations* in the image ($\hat{t}$ and $\hat{n}$).	τ is precisely computable from divergence *only along the optical axis*, (unless surface is fronto-parallel).
τ is precisely computable from divergence along the direction of motion *regardless of surface orientation, or location in the image.*	τ is precisely computable from divergence along the direction of motion *only if it is aligned with the optical axis or surface is fronto-parallel.*

Table 6.1: Comparison of constraints on time-to-contact estimation under spherical and perspective projection.

The advantages highlighted in Table 6.1 are significant in the context of designing velocity control schemes based on time-to-contact estimates. Given the reduced assumptions under a spherical projection framework, we may expect to obtain more robust and accurate estimates of time-to-contact. Note also that there is no requirement for camera calibration (except if re-mapping to a spherical model).

6.2.2 The point of maximum divergence

For a unit view sphere approaching an infinitely large planar surface, let θ_{nt} be the angle subtending the arc connecting the direction of translation, $\hat{t}$, and the direction of the surface normal, $\hat{n}$ (*i.e.,* the angle of approach). For an arbitrary direction, $\hat{p}$ on the view sphere, let θ_{pt} and θ_{pn} be the angles subtending the arc connecting $\hat{t}$ and $\hat{p}$, and $\hat{n}$ and $\hat{p}$ respectively. From these definitions we propose the following theorem:

Theorem 6.2.1. *Assuming $\theta_{nt} \in [0, \frac{\pi}{2}]$, if $\theta_{pt} = \theta_{pn}$ and $\theta_{pt} + \theta_{pn} = \theta_{nt}$, then $\hat{p}$ must*

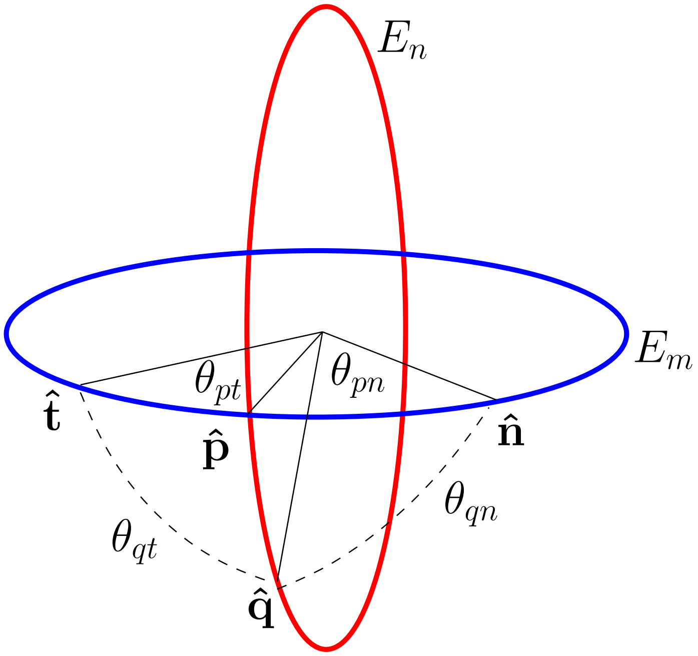

Figure 6.1: Geometric framework for proof of the max-div property.

be the point of maximum divergence on the view sphere. That is, for a planar surface projecting onto the full arc connecting $\hat{t}$ and $\hat{n}$, the point of maximum divergence is always located halfway between $\hat{t}$ and $\hat{n}$ on this arc.

Proof. We first show that the point of maximum divergence must occur on the great circle, $E_{\text{m}} \in \mathbb{R}^3$ passing through $\hat{n}$ and $\hat{t}$. Let $\hat{p}$ be constrained to E_{m}. Let $E_n(\hat{p})$ define the orthogonal great circle to E_{m} at $\hat{p}$. Let $\hat{q} \in \mathbb{R}^3$ be any point on $E_n(\hat{p})$, such that $\hat{q} \neq \hat{p}$ (see Figure 6.1. We seek to show that for any $\hat{p} \neq \hat{q}$, $\text{div}(\hat{p}) > \text{div}(\hat{q})$. The proof is by contradiction.

Re-writing Equation 6.3 in angular form, we obtain:

$$\text{div}(\hat{p}) = \frac{v}{R_o}\Big(2\cos(\theta_{pn})\cos(\theta_{pt}) - \cos(\theta_{nt})\Big), \tag{6.6}$$

from which we express the inequality $\text{div}(\hat{p}) > \text{div}(\hat{q})$, such that:

$$\cos(\theta_{\text{pn}})\cos(\theta_{\text{pt}}) \leq \cos(\theta_{\text{qn}})\cos(\theta_{\text{qt}}), \tag{6.7}$$

where θ_{qt} and θ_{qn} are angles subtending the arcs connecting $\hat{q}$ and $\hat{t}$, and $\hat{q}$ and $\hat{n}$ respectively. Further re-arrangement yields:

$$\frac{\cos(\theta_{\mathrm{pn}})}{\cos(\theta_{\mathrm{qn}})} \leq \frac{\cos(\theta_{\mathrm{qt}})}{\cos(\theta_{\mathrm{pt}})}. \tag{6.8}$$

Note that $\hat{q}$ is located on an orthogonal great circle to E_{m}. Thus, $\hat{q}$ is always at a greater distance from $\hat{t}$, and $\hat{n}$, than $\hat{p}$ which lies on the shortest arc connecting $\hat{t}$ and $\hat{n}$. Thus, $\theta_{pt} < \theta_{qt}$ and $\theta_{pn} < \theta_{qn}$, from which we infer:

$$\frac{\cos(\theta_{\mathrm{pn}})}{\cos(\theta_{\mathrm{qn}})} > 1, \text{and} \frac{\cos(\theta_{\mathrm{qt}})}{\cos(\theta_{\mathrm{pt}})} < 1. \tag{6.9}$$

It therefore follows that:

$$\frac{\cos(\theta_{\mathrm{pn}})}{\cos(\theta_{\mathrm{qn}})} > \frac{\cos(\theta_{\mathrm{qt}})}{\cos(\theta_{\mathrm{pt}})}, \tag{6.10}$$

thus contradicting the original inequality given in Equation 6.8. It therefore follows that $\mathrm{div}(\hat{p}) > \mathrm{div}(\hat{q})$ for all $\hat{p}$ and $\hat{q}$, and thus the point of maximum divergence must occur on the great circle E_m.

We now prove that the point of maximum divergence occurs halfway along the arc connecting $\hat{t}$ and $\hat{n}$. Considering Equation 6.6 again, we constrain $\hat{p}$ to locations on the great circle passing through $\hat{t}$ and $\hat{n}$, and exploit the following relationship:

$$\theta_{\mathrm{pn}} = \theta_{\mathrm{nt}} - \theta_{\mathrm{pt}}. \tag{6.11}$$

From this we may rewrite Equation 6.6 as:

$$\mathrm{div}(\theta_{\mathrm{pt}}) = \frac{v}{R_o}\Big(2\cos(\theta_{\mathrm{nt}} - \theta_{\mathrm{pt}})\cos(\theta_{pt}) - \cos(\theta_{nt})\Big). \tag{6.12}$$

We therefore seek to find the angle, θ_{pt}, that maximises this equation. Taking its

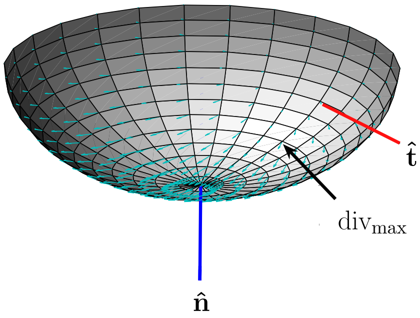

Figure 6.2: Example of max-div property on the view sphere. Increasing brightness on the sphere surface indicates the increasing divergence.

derivative and solving for 0 we obtain:

$$\begin{aligned} 0 &= \frac{d}{\theta_{\mathrm{pt}}}\Big(2\cos(\theta_{\mathrm{nt}}-\theta_{\mathrm{pt}})\cos(\theta_{\mathrm{pt}})-\cos(\theta_{\mathrm{nt}})\Big), \\ &= \cos(\theta_{pt})\sin(\theta_{nt}-\theta_{pt})-\sin(\theta_{pt})\cos(\theta_{nt}-\theta_{pt}), \\ &= 2\sin(\theta_{\mathrm{nt}}-2\theta_{\mathrm{pt}}). \end{aligned}$$

Therefore:

$$\begin{aligned} 0 &= \theta_{\mathrm{nt}}-2\theta_{\mathrm{pt}}, \\ \theta_{\mathrm{pt}} &= \frac{\theta_{\mathrm{nt}}}{2}. \end{aligned} \tag{6.13}$$

Noting that $\theta_{\mathrm{pn}} = \theta_{\mathrm{nt}} - \theta_{\mathrm{pt}}$, it also follows that $\theta_{\mathrm{pn}} = \frac{\theta_{\mathrm{nt}}}{2}$. Thus, we have proven that the point of maximum divergence must occur halfway along the arc connecting $\hat{t}$ and $\hat{n}$. □

Figure 6.2 shows an example of the max-div property, as computed from simulation.

6.2.2.1 What does max-div represent?

Substituting θ_{pt} for $\frac{1}{2}\theta_{\text{nt}}$ in Equation 6.12, we obtain the following expression for the maximum divergence quantity:

$$\text{div}_{\text{max}} = -\frac{v}{R_o}\left(2\cos^2\left(\frac{\theta_{\text{nt}}}{2}\right) - \cos(\theta_{nt})\right). \qquad (6.14)$$

Applying the trigonometric identity $2\cos^2(\theta) - \cos(2\theta) = 1$, and taking its recipricol, the expression becomes:

$$\frac{1}{\text{div}_{\text{max}}} = -\frac{R_o}{v}. \qquad (6.15)$$

The above equation defines the scaled depth of the surface along $\hat{n}$. Most significantly, this quantity is obtainable directly from the measured maximum divergence, requiring no knowledge of the surface orientation, the direction of egomotion, or the direction of the surface normal.

6.3 Unifying landing and docking using max-div

The max-div property gives rise to a divergence-based control scheme for landing and docking with a planar surface of arbitrary orientation. Moreover, this property offers a means of regulating both the velocity of the vehicle, and its approach angle with respect to a planar surface. Below we detail proposed control schemes that achieve both these capabilities through the use of the max-div point.

6.3.1 Regulating approach velocity

In Chapter 5, we demonstrated that velocity towards near fronto-parallel surfaces may be controlled by maintaining constant divergence during the approach. Equation 6.15 constitutes a general solution to the docking/landing problem as it may be applied to regulate velocity for any angle of approach, without alteration. Moreover, Equation 6.15 shows that the maximum divergence value is invariant to the angle of approach, and is therefore equally suitable for both frontal and grazing approaches. Thus, we may apply the same control law as proposed in Chapter 5 (Equation 5.25).

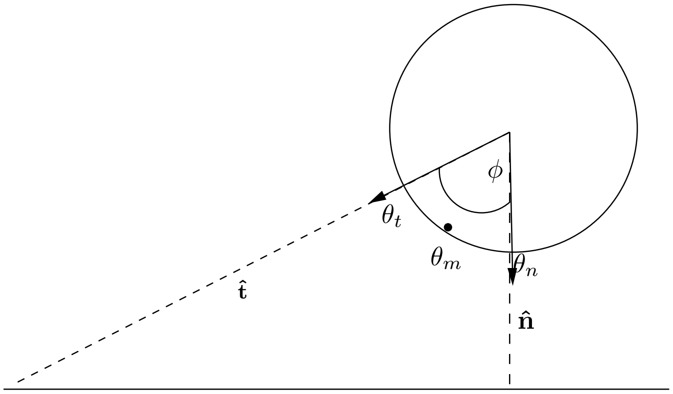

Figure 6.3: Geometric model for approach angle control on view sphere

6.3.2 Regulating approach angle

The relationship between the point of maximum divergence, the direction of translation and the planar surface normal gives rise to a simple visual servoing strategy for maintaining a given orientation with respect to a planar surface. This can be achieved without explicit tracking of the direction of ego-motion and is invariant to rotational flow.

We consider only regulation of the approach angle, θ_{nt}, defined as the angle between the sphere's direction of translation $\hat{t}$, and the direction of the surface normal, $\hat{n}$.

Let θ_t, $\theta_n \in [0, 2\pi]$ be the angular location of $\hat{t}$ and $\hat{n}$ respectively, on the great circle, E_m, passing through both points. Let $\hat{m}$ be the direction of the max-div point on the great circle E_m, and θ_m, its angular location on E_m, as shown in Figure 6.3. Given a stationary ground plane, we assume θ_n to be fixed during the approach, and thus any change in θ_{nt} to be due only to changes of θ_t. This constraint ensures the location of the maximum divergence point represents a unique angle of approach with respect to the surface.

Let θ_m be the location of the point of maximum divergence on E_m. From Theorem 6.2.1, we note that $\theta_m = \frac{1}{2}(\theta_t + \theta_n)$. Given θ_n is stationary, we may assume that changes in θ_m ($\Delta\theta_m$) will be due only due to changes in θ_t ($\Delta\theta_t$), such that:

$$\Delta_{\theta_m} = \frac{1}{2}\Delta_{\theta_t} \tag{6.16}$$

Therefore, maintaining θ_{nt} during an approach towards a planar surface can be achieved through proportional adjustments of θ_t, such that θ_m remains at a constant location on E_m.

This gives rise to a simple control law for controlling the approach angle within the plane of E_m:

$$\theta_t(t) = \theta_t(t-1) - 2(\theta_s - \theta_m(t)), \tag{6.17}$$

where θ_s is the set point for the location of the maximum divergence point, and $\theta_m(t)$ is the computed location of the maximum divergence point at time t.

It is important to note that (6.17) does not consider lateral motion of the maximum divergence point (*i.e.,* when $\hat{m}$ does not lie on E_m). To maintain the sphere's orientation with respect to the ground plane, two rotations are required. One to bring $\hat{m}$ back to E_m, and another to adjust the angle of approach as defined in Equation (6.17). Given, for example, knowledge of the direction of the ground plane, then both the angle of approach, and the orientation of the sphere with respect to the ground plane can be maintained[1]. Note, however, that the control scheme places no constraints on the rotation of the sphere to maintain the angle of approach. Thus, the rotational axes used to re-align $\hat{m}$ with the reference location may be chosen arbitrarily.

6.4 Open-loop performance evaluation

In the next two sections we present experiments to validate and test the proposed max-div landing/docking scheme. In this section, we present two sets of open-loop experiments, over pre-constructed real image sequences. In these tests, we seek to examine the underlying time-to-contact estimation, and the feasibility of tracking the max-div point for direction control.

6.4.1 Omni-directional cameras employed

All experiments presented (both open and closed-loop) employ one of two cameras providing hemispherical projections of the scene. We describe both cameras below and

[1] such information may be available from observations of the horizon, or gravity sensors.

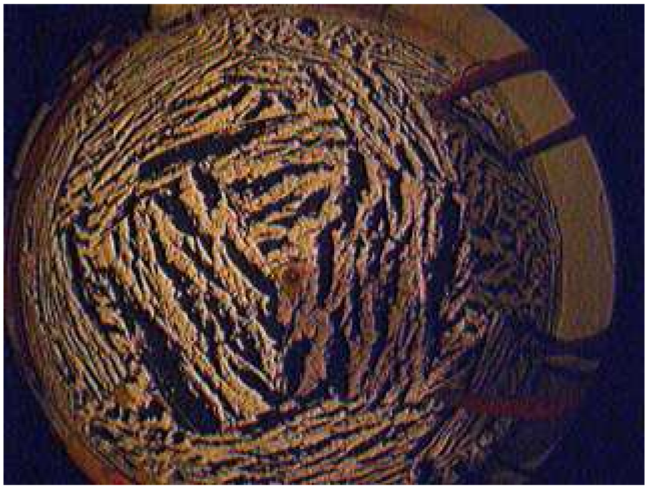

Figure 6.4: Sample frame from indoor landing sequences. For each sequence, the camera was incrementally lowered towards the ground plane at a preset angle of approach.

simply refer to the appropriate camera by the specified name in subsequent experiment descriptions.

Omni-tech Robotics Unibrain Fire-i BCL 1.2 lens provides an approximately spherical projection over a 190^o field of view, The image is projected onto a standard CCD sensor array. The central region of the visual field provides a close approximation to a spherical projection, and is thus deemed suitable for use in testing the time-to-contact estimator. A mapping of each pixel to the unit sphere was obtained from calibration using Kannala and Brandt's [2006] method. We capture images from the camera at a resolution of 320 × 240 pixels. We refer to this camera as the *omnitech* camera.

Point Grey Research Ladybug camera provides an almost global view of the scene from six mounted cameras within the device. In this work, we make use of inbuilt image stitching which provides a 180^o domed view of the scene. We capture images of this domed view at a resolution of 512 × 512 pixels. We refer to this camera as the *ladybug camera*.

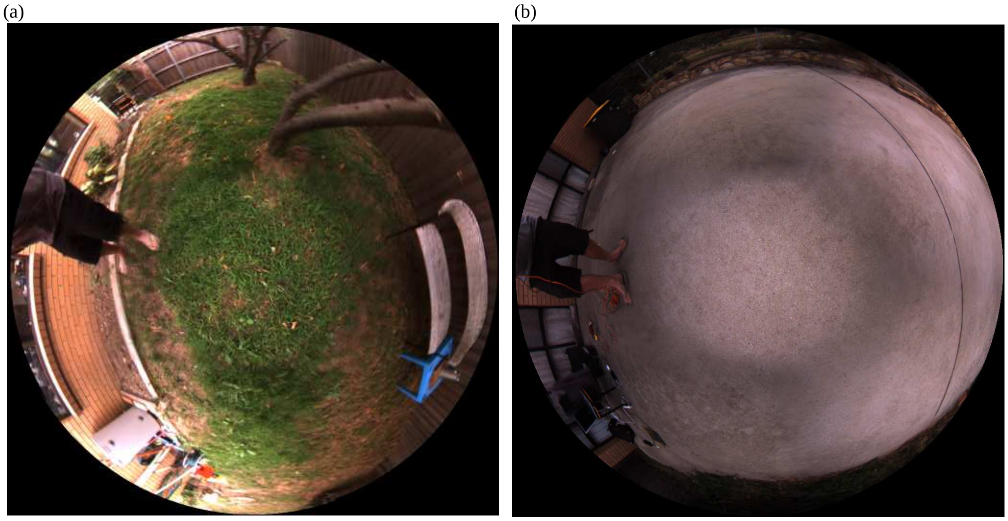

Figure 6.5: Sample frames taken from: (a) the grass-landing sequences, and, (b) the cement-landing sequence.

6.4.2 Indoor image sequences (controlled environment and motion)

Using the omnitech camera, four indoor image sequences were constructed, each depicting the motion of the camera towards a heavily textured, planar surface. The four sequences depict discrete angles of approach towards the surface: 0^o (*i.e.,* a frontal approach), 22.5^o, 45^o and 67.5^o. In all sequences, the camera was positioned such that the image centre was approximately aligned with the surface normal. Figure 6.4 shows a sample frame from one of the indoor landing sequences. The camera's velocity was 5mm per frame in the direction of approach for each trial. Note that while conditions were highly controlled, the camera's motion was subject to errors in its alignment with the surface due to imprecision in camera movement which was performed by hand.

Using these image sequences, we assess the feasibility of applying the proposed control scheme. For velocity control, we consider the accuracy and stability of time-to-contact estimates taken from the measured divergence at the max-div point. For directional control, the validity of the relationship between the max-div point and the approach angle is examined. The estimated location of the max-div point is recorded for each frame and mapped to the unit view sphere from calibration data. The angular displacement of the max-div point with respect to the image centre is then computed.

6.4.3 Outdoor image sequences - hand-held camera in uncontrolled environments

Outdoor image sequences were constructed using the ladybug camera. The camera was mounted on a tripod which was then hand-held horizontally, and extended out in front of the body, pointing towards the ground. In all sequences, the camera was walked through the environment, while simultaneously being lowered towards the ground by hand. Three outdoor sequences were constructed:

1. the *straight grass-landing sequence*, depicting a predominantly straight and gradual descent towards an unevenly grassed ground plane;

2. the *rotating grass-landing sequence*, depicting the camera's descent towards the same grassed surface, while simultaneously being swept across the scene from right to left; and,

3. the *cement-landing sequence*, depicting a predominantly straight and more rapid descent towards a significantly less textured, smooth cement surface.

Sample frames from each sequence are shown in Figure 6.5.

In all sequences, the camera's velocity and trajectory are approximately constant, though the less precise conditions of their construction unavoidably introduce variations. To avoid the influence of other objects in the periphery of the scene, divergence was measured only within a ninety pixel radius of the image centre. Assuming the surface normal is in the approximate direction of the image centre, this region encompasses the range of possible max-div locations for any descent angle.

6.4.3.1 Optical flow and divergence estimation

In all experiments (open and closed-loop), optical flow was computed for every eighth pixel using a pyramidal implementation of Lucas and Kanade's gradient-based technique (discussed in Chapter 3, Section 3.2.2.5). Divergence was computed using a linear size five Sobel kernel for gradient estimates, providing divergence estimates for each defined flow vector location. Time-to-contact estimates were obtained from the recipricol of the average of divergence estimates within a 5×5 vector patch, centred

approach angle (deg)	max div loc (deg)	std dev (deg)
0^o	5.8^o	1.5^o
22.5^o	16.7^o	2.5^o
45^o	19.6^o	8.6^o
67.5^o	31.3^o	15.2^o

Table 6.2: Results for indoor landing sequences showing average radial distance from image centre of max-div point

on the estimated point of maximum divergence. The divergence maximum was defined simply as the location associated with the highest divergence in the image.

6.4.4 Open-loop results

6.4.4.1 Indoor landing sequence results

Figures 6.6 through 6.9 show sample frames from each indoor sequence, showing also the estimated optical flow field, the location of the max-div point (indicated by the white cross), and the divergence as measured across the image. Video footage showing results for the 0^o and 45^o landing sequences is provided on the thesis CD-ROM (Videos 6.1a and 6.1b). Figure 6.10 shows time-to-contact estimates obtained at the point of maximum divergence for each of the four approach angles. To account for scaling differences between the estimated time-to-contact and ground truth, time and time-to-contact have been normalised using the following normalisation equation:

$$\hat{\tau}_{\text{i}} = \frac{\tau_i - \tau_{\text{max}}}{\tau_{\text{max}} - \tau_{\text{min}}},$$

where $\hat{\tau}_i$ is the normalisation of τ_i from the trial, and τ_{max} and τ_{min} are the maximum and minimum time-to-contact estimates from the trial respectively. Ground truth, after normalisation, is the line passing through $\tau = 1$ and $t = 1$.

In general, time-to-contact estimates remain stable, and provide a close match with ground truth. Accuracy and stability degrades as the angle of approach increases, as is evident for the 67.5^o sequence in Figure 6.10(d). In all cases, however, fluctuations appear to diminish as time-to-contact decreases.

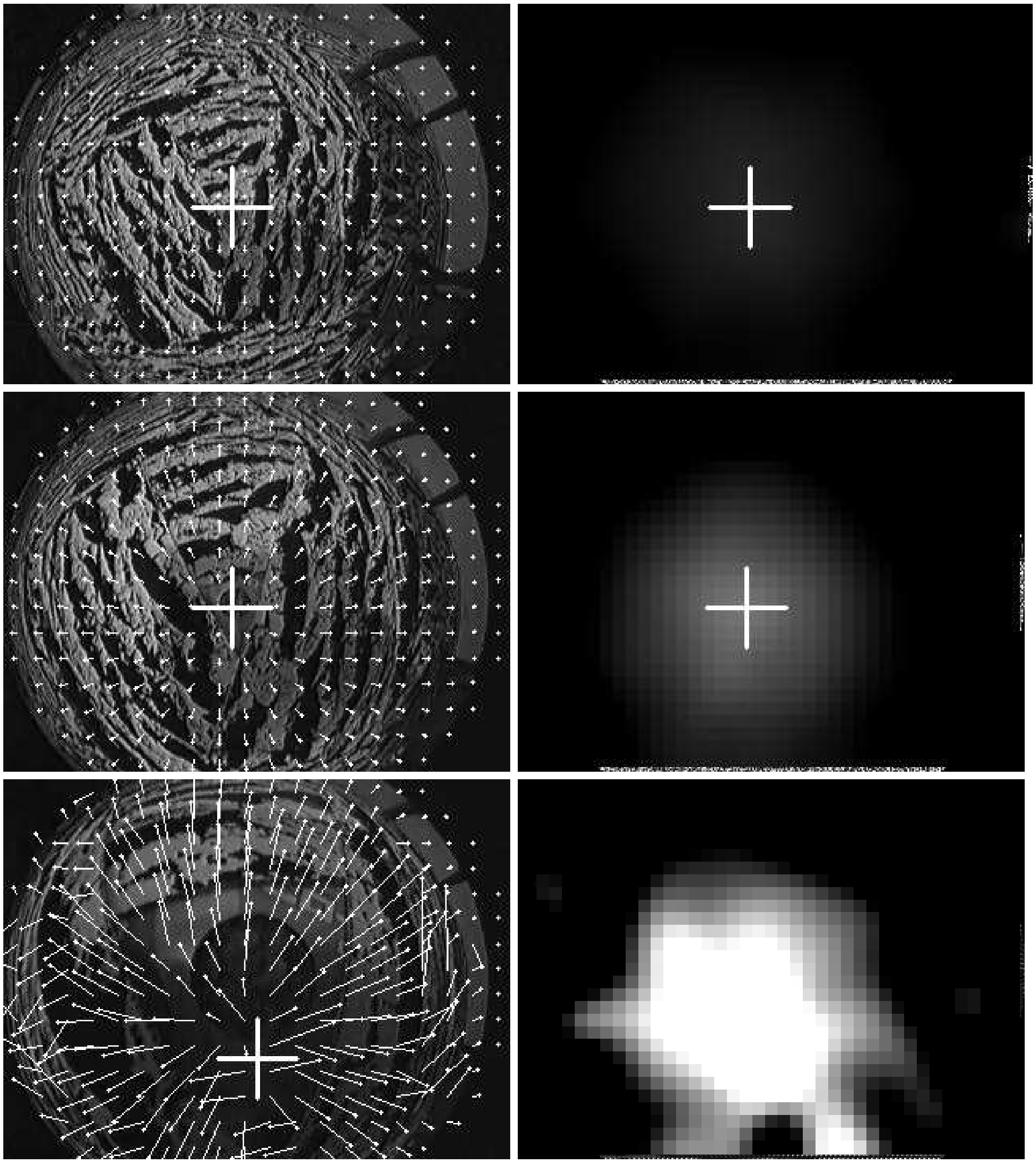

Figure 6.6: Sample frames from indoor landing sequence for 0^o approach. Also shown is the estimated optical flow field, and the estimated location of the max-div point (white cross). Right column shows divergence maps estimated for each corresponding frame. Greater intensity represents larger divergence.

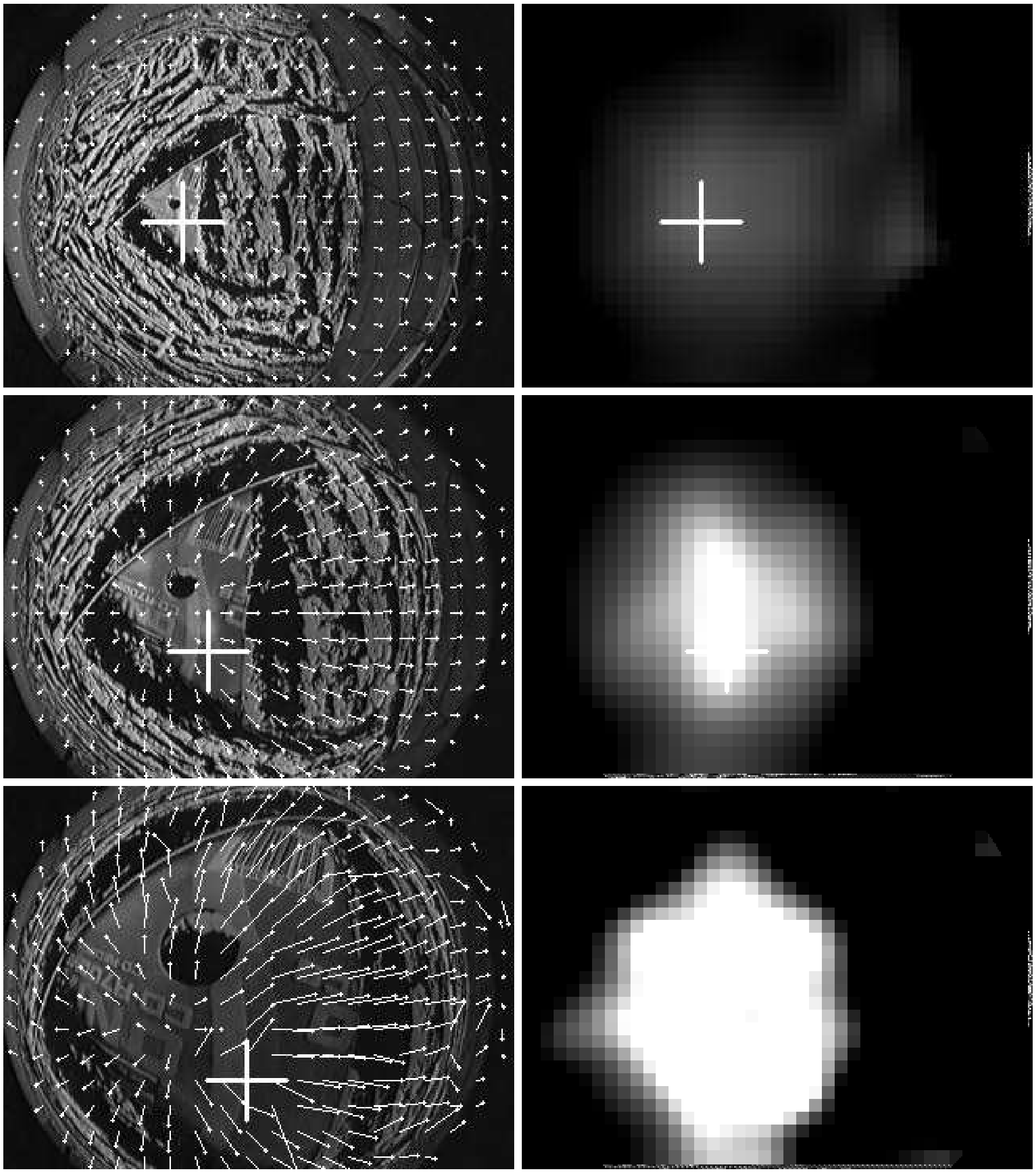

Figure 6.7: Sample frames from indoor landing sequence for 22.5^o approach. Also shown is the estimated optical flow field, and the estimated location of the max-div point (white cross). Right column shows divergence maps estimated for each corresponding frame. Greater intensity represents larger divergence.

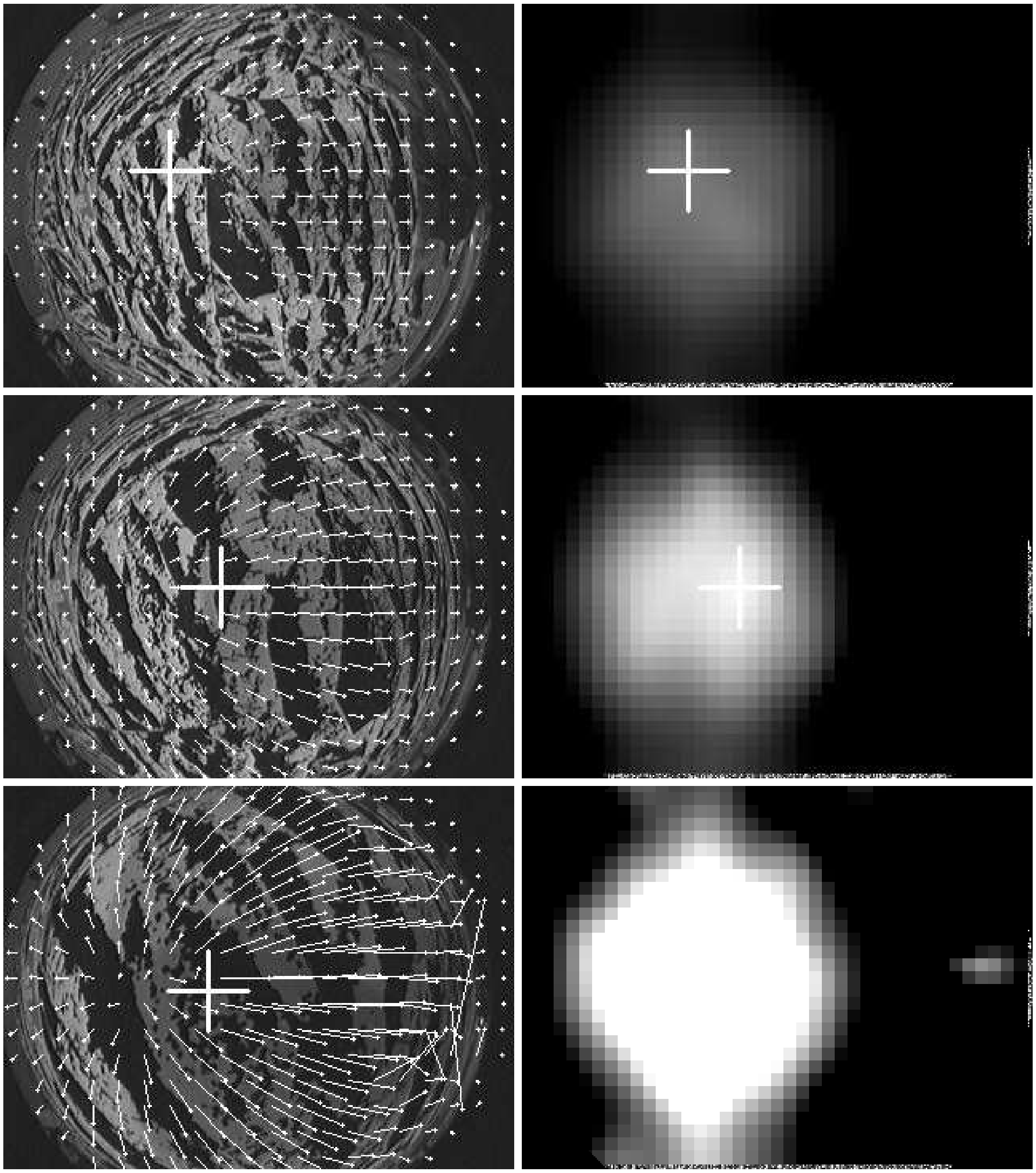

Figure 6.8: Sample frames from indoor landing sequence for 45^o approach. Also shown is the estimated optical flow field, and the estimated location of the max-div point (white cross). Right column shows divergence maps estimated for each corresponding frame. Greater intensity represents larger divergence.

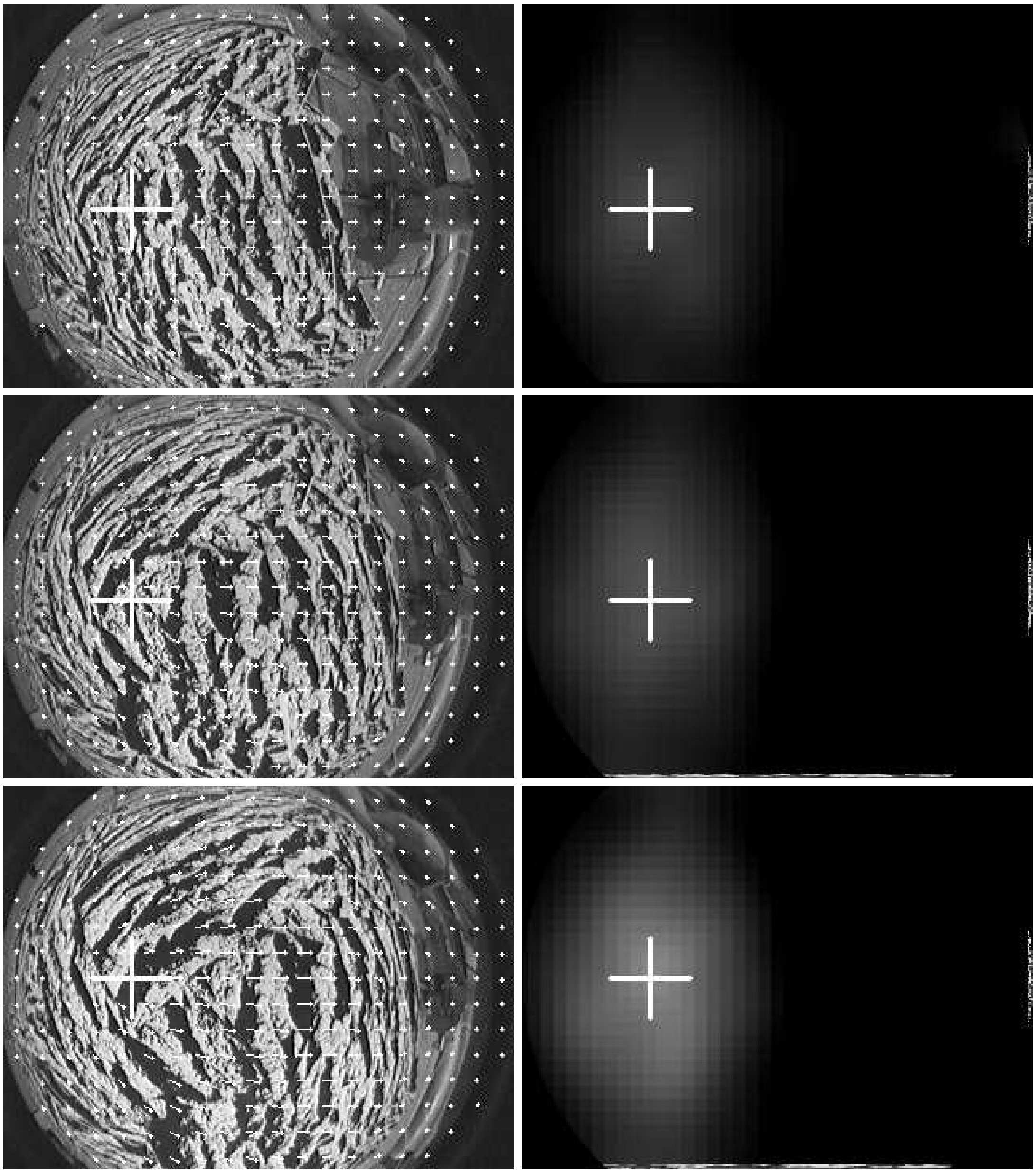

Figure 6.9: Sample frames from indoor landing sequence for 67.5^o approach. Also shown is the estimated optical flow field, and the estimated location of the max-div point (white cross). Right column shows divergence maps estimated for each corresponding frame. Greater intensity represents larger divergence.

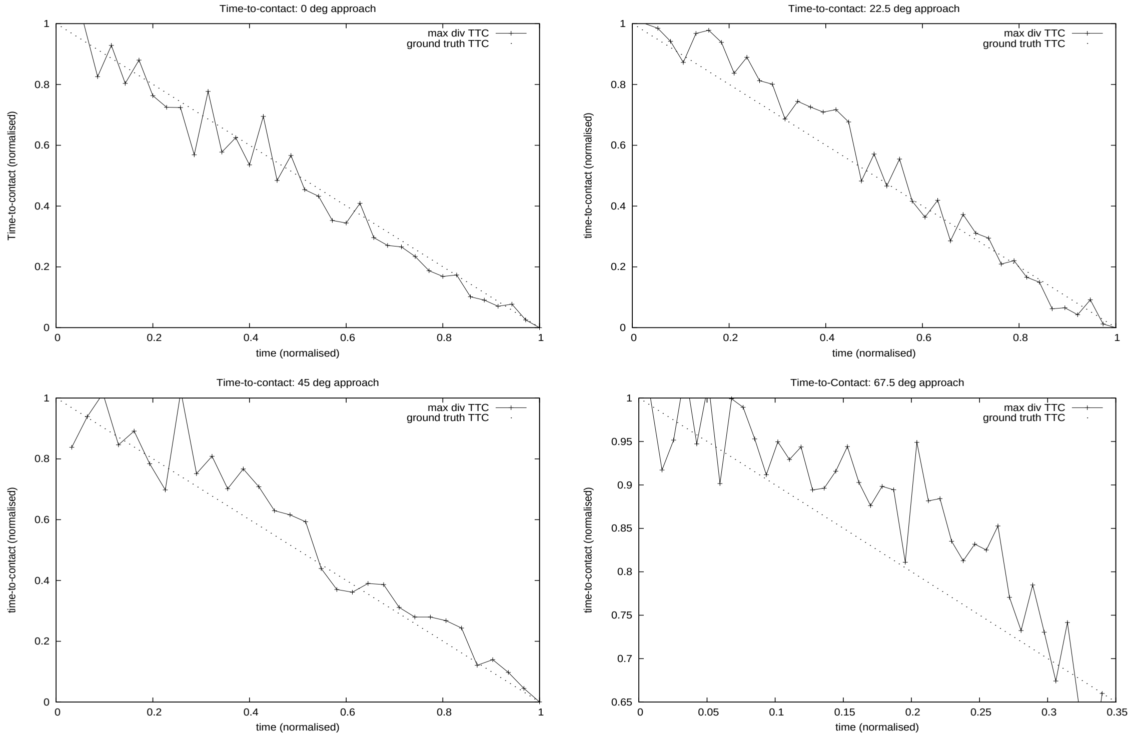

Figure 6.10: Time-to-contact results for open-loop, indoor landing sequences using the max-div estimation scheme. Ground truth (the dotted line) is also given for each angle of approach.

Table 6.2 shows the average angular displacement of the max-div point with respect to the image centre for each sequence, as well as the standard deviation of the estimated angle. Results appear consistent with theoretical expectations, showing average angular displacements of the max-div point to be approximately half the angle of approach. While standard deviations in the estimated location of the max-div point increase with the angle of approach, mean max-div locations (*i.e.,* the direction of the surface normal) still provide an accurate gauge of the approach angle.

6.4.4.2 Outdoor landing sequence results

Figures 6.11, 6.12 and 6.13 show the computed optical flow field, including the estimated location of the max-div point, and grayscale images representing divergence levels computed from the flow field, for each outdoor image sequence. Increasing brightness indicates higher divergence (*i.e.,* decreasing time-to-contact). Video footage of these results for both grass landing sequences is provided on the thesis CD-ROM (Videos 6.2a and 6.2b). Sample frames are approximately one second apart. Distinct regions of positive divergence around the estimated max-div point are evident in the divergence maps of all sequences.

Figure 6.14 shows time-to-contact estimates obtained at the max-div point for the last 30 frames of each sequence. As in indoor image sequence experiments, results are normalised. As expected, all outdoor sequences exhibit increased noise levels, and less conformity with ground truth compared with indoor experiments. Closer inspection of time-to-contact estimates in regions of deviation from ground truth shows, however, that local temporal consistency is generally still apparent. This suggests that such deviations are not likely a result of erroneous time-to-contact estimates, but rather a response to genuine changes in the hand-held camera's motion during the approach. This is particularly evident in Figure 6.14(c) between $t = 0.3$ and $t = 0.7$, but also between $t = 0.2$ and $t = 0.45$, and $t = 0.6$ and $t = 1$ in Figure 6.14(a), and between $t = 0.2$ and $t = 0.5$ in Figure 6.14(b). Instances of large fluctuations are rare (only once in each sequence). Notably, consistency in time-to-contact estimation appears to improve over the smoother, less textured cement surface than the grassed surface. The inclusion of camera rotation in the *rotating grass-landing sequence* does not appear

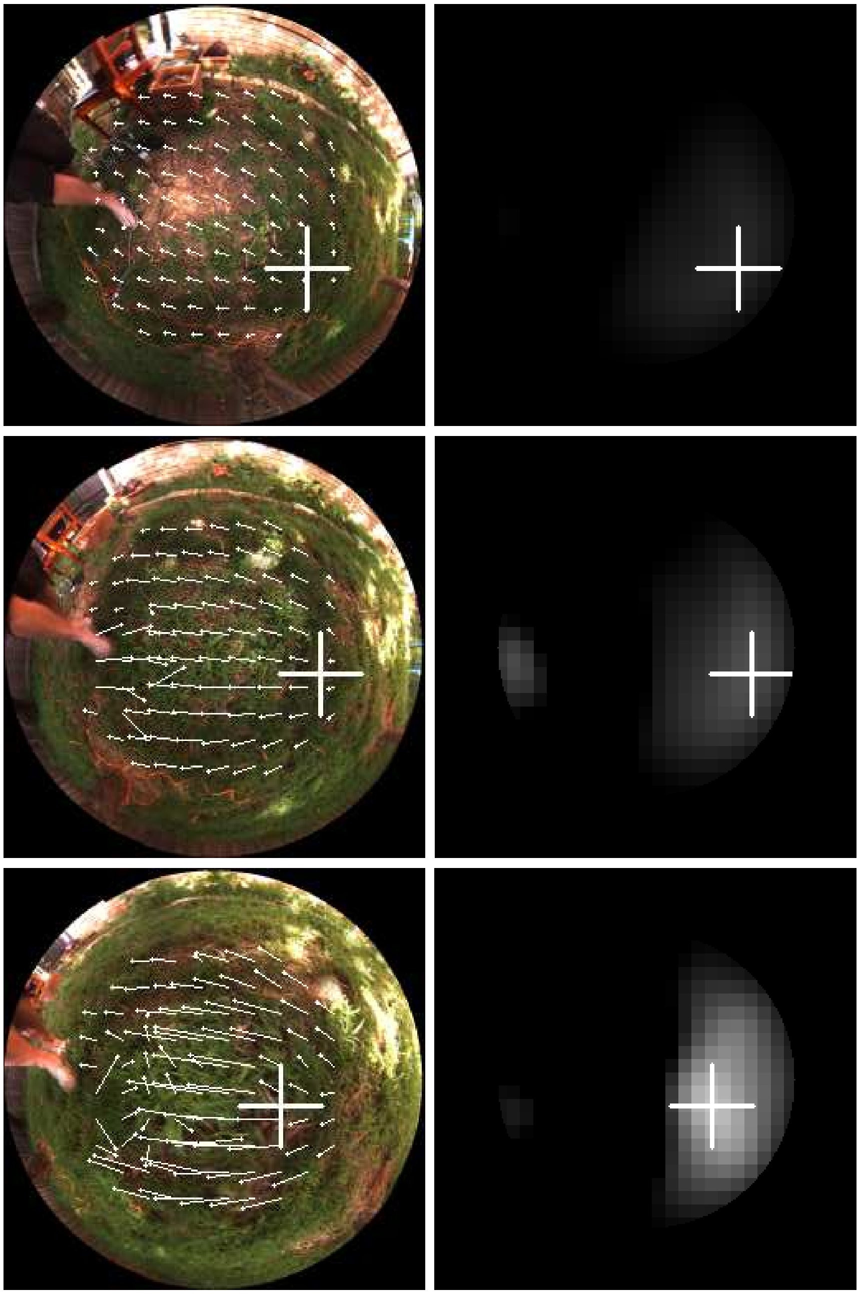

Figure 6.11: Sample frames from open-loop, outdoor grass landing sequence (straight approach). The left column for each shows the estimated flow field, and the estimated location of the max-div point (white cross). Right column shows divergence maps estimated for each corresponding frame. Greater intensity represents larger divergence.

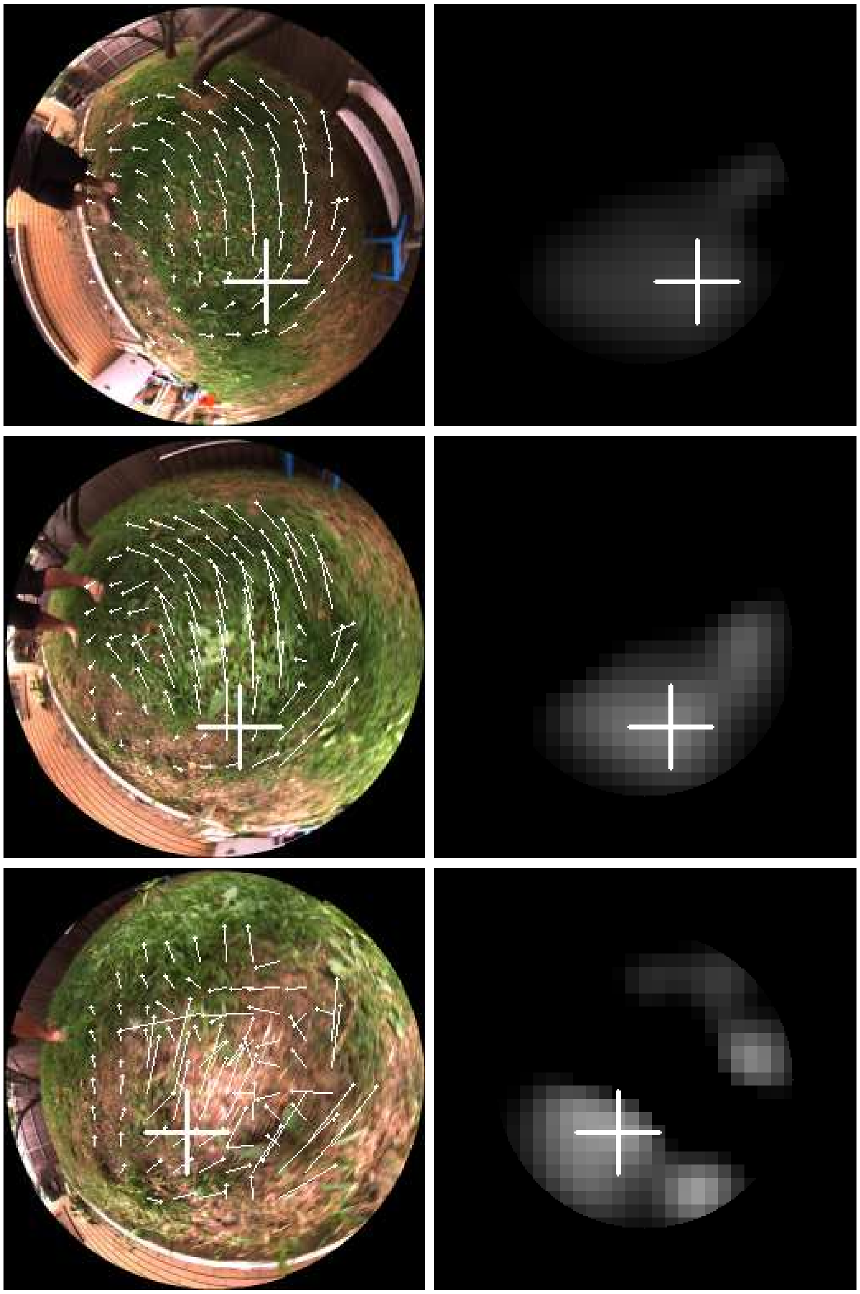

Figure 6.12: Sample frames from open-loop, outdoor grass-landing sequence (rotating approach). The left column for each shows the estimated flow field, and the estimated location of the max-div point (white cross). Right column shows divergence maps estimated for each corresponding frame. Greater intensity represents larger divergence.

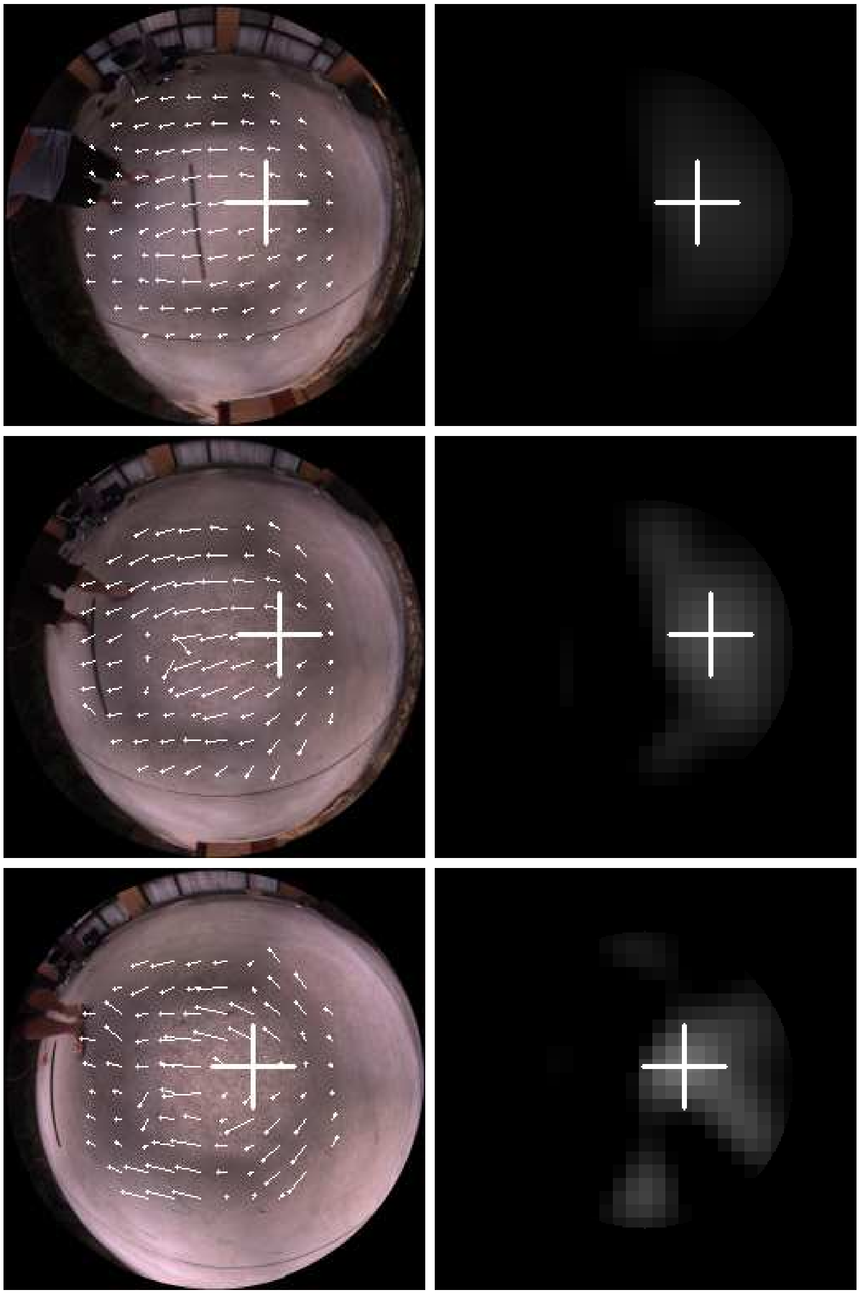

Figure 6.13: Sample frames from open-loop, outdoor cement landing sequence. The left column for each shows the estimated flow field, and the estimated location of the max-div point (white cross). Right column shows divergence maps estimated for each corresponding frame. Greater intensity represents larger divergence.

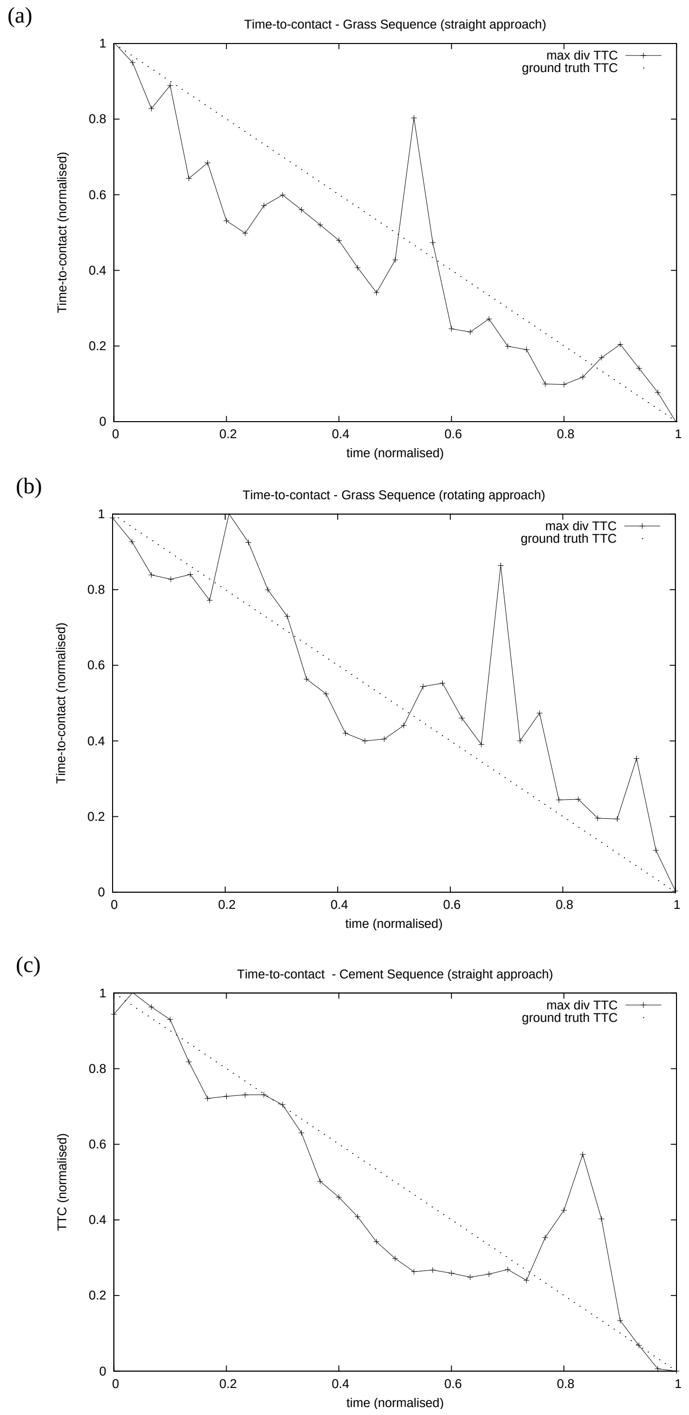

Figure 6.14: Outdoor open-loop time-to-contact results for: (a) grass-landing sequence (straight), (b) grass-landing sequence (rotate), and (c) cement-landing sequence Results are shown for the last 30 frames of each sequence. Note that camera was hand-held during the construction of each sequence.

Sequence	max div pos (mean and std dev)	estimated appr angle (2× max div pos)
grass (straight)	37.8^o, $\sigma = 8.6^o$	75.6^o
cement	21.1^o, $\sigma = 13.5^o$	40.8^o

Table 6.3: Approach angle data for straight outdoor landing sequences.

to significantly effect time-to-contact estimates compared with the straight motion approach. Most significantly, the linear downward trend of time-to-contact estimates is preserved across all sequences.

Table 6.3 shows the estimated location of the max-div point on the view sphere with respect to the image centre (the approximate direction of the surface normal) for the straight grass-landing and cement-landing sequences. Given the imprecise nature of the experiments, it is difficult to quantitatively assess the accuracy of the estimated approach angles. However, the estimated approach angles clearly distinguish the sharper descent angle of the cement-landing experiment from the shallower descent of the grass-landing sequence.

6.4.5 Open-loop results discussion

Open-loop testing results appear to support the viability of the max-div docking scheme, and support the generality of the proposed time-to-contact estimator for any angle of approach. While some loss of consistency in time-to-contact estimation is apparent in the 67.5^o indoor sequences, this may be due to the camera's velocity being small with respect to the distance. Closed-loop trials will provide further insight. Notably, workable time-to-contact estimates are obtained at similar descent angles in outdoor image sequences.

Some loss of consistency in time-to-contact estimation is apparent in outdoor experiments. However, reduced surface texture, and the inclusion of camera rotation do not appear to adversely effect time-to-contact estimation performance, suggesting the unevenness of the grassed-surface has the most impact on divergence estimation. Discontinuities on the surface were observed to occasionally give rise to false max-div estimates. While the pyramidal estimation of the optical flow field provides some robustness to this, it was observed that robustness could be improved further by smoothing divergence estimates before estimating the location of the max-div point. Most significantly, however, a linear downward trend in time-to-contact estimates is retained across all sequences, providing strong support for the feasibility of the max-div control scheme.

6.5 Closed-loop performance evaluation

In this section we consider the performance of the max-div control scheme in the context of a closed-loop control system. Specifically, we assess the control scheme's ability to robustly perform controlled approaches towards a planar surface of arbitrary orientation. We first describe the experiments conducted, before presenting results.

6.5.1 Closed-loop simulation testing

A simulation was constructed depicting the motion of a vehicle in 3D space towards an infinitely large planar surface. To control the vehicle's velocity and angle of approach, the max-div control scheme was implemented for use in the vehicle's control loop. The vehicle model was assumed to be equipped with a spherical vision sensor of unit radius, providing a spherical projection of the ground plane. Optical flow vectors induced by the vehicle's motion with respect to the ground plane were computed from the vehicle's known motion parameters, and its distance from the surface. Flow field divergence was then computed from discrete samples of the flow field on the local tangent plane about the given region of interest.

Trials were conducted for angles of approach ranging from 0^o (a frontal approach), to 67.5^o with respect to the surface normal. In each trial, the max-div point was used to maintain the desired angle, while also regulating the vehicle's velocity to maintain a constant maximum divergence value. To simulate real-world conditions, Gaussian noise was added to the intended direction of motion, and velocity, thus forcing correctional adjustments of heading and speed at each iteration of the control loop.

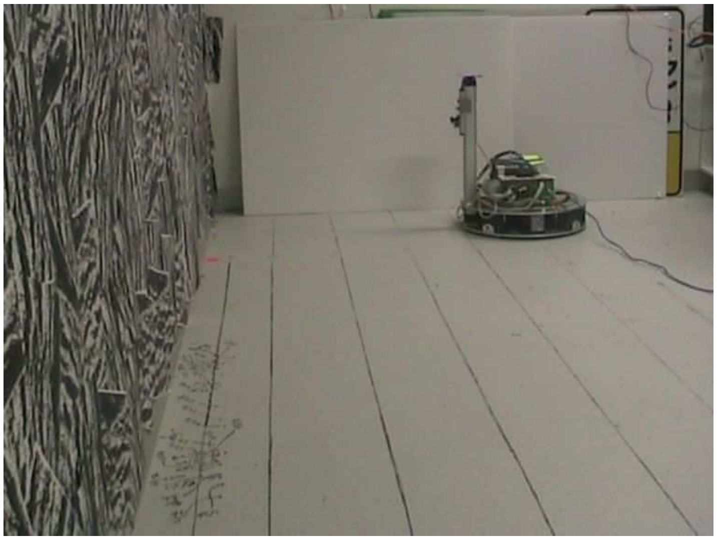

Figure 6.15: Side view of on-board experimental workspace, and robot, used for closed-loop on-board trials of the max-div docking/landing control scheme.

6.5.2 Mobile robot landing/docking experiments

The max-div control scheme was implemented for closed-loop control of a mobile robot. For comparison, the *graze-landing model* proposed by Srinivasan *et al.* [2000] was also implemented for closed-loop control of the robot.

The omnitech camera (described in Section 6.4.1 was attached to the front of the robot, providing a forward-facing, approximately hemispherical field of view. The robot drive system provided omni-directional motion on the ground plane, thus allowing instantaneous translation in any direction.

Across all trials, the robot was required to perform a controlled approach towards an upright surface, attempting to stop as close as possible to the surface without collision. As in previous experiments, trials were conducted for a range of approach angles: 0^o, 22.5^o, 45^o, and 67.5^o. Five trials were conducted for each angle of approach. For all trials, the robot's initial direction of motion was parallel to the surface, thus requiring explicit adjustments of direction to achieve the desired approach angle. The robot's initial pose was facing the docking surface, at a distance of 100 cm for 67.5^o trials, and 125 cm for all others[2]. In each trial, the robot's path on the ground plane

[2] Physical space constraints forced 67.5^o trials to be started closer to the docking surface.

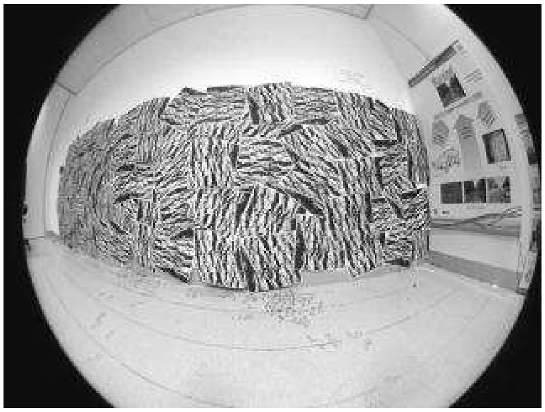

Figure 6.16: View from onboard camera of closed-loop on-board experimental workspace.

was plotted via tracking software and footage captured from a calibrated overhead camera. The robot's initial velocity was set to $20cms^{-1}$. Figure 6.15 shows the robot, and the experimental workspace used for on-board landing/docking trials. Figure 6.16 shows the view from the onboard camera. We discuss the implementation of both the max-div, and graze-landing strategies below.

6.5.2.1 Implementation of max-div control scheme

Optical flow and divergence estimation were computed as in indoor open-loop experiments, however, the field of view was restricted to the central third strip of the image to avoid competing divergence maximum points from other surfaces such as the ground plane. Velocity control was achieved via a simple Proportional-Derivative (PD) control law. The discrete time realisation of the control law applied is:

$$v(t) = \Delta v(t-1) + \frac{\Delta}{m}\Big(K_p(D_{\text{ref}} - D_t) - K_d(D_t - D_{t-1})\Big), \tag{6.18}$$

where $v(t)$ is the velocity control input at time t, Δ is the discretisation time, m is a virtual vehicle mass, D_t is the estimated max-div value at time t, D_{ref} is the reference max-div location set-point, and K_p and K_d are proportional and derivative control gains respectively. A proof of stability for divergence-based velocity control is provided in [McCarthy et al. 2008]. Note that the stability proof contained within this paper is not a contribution of the author of this thesis.

Directional control was constrained to angular adjustments of the translation direction on the ground plane. This was achieved using a variant of the proposed control law defined in Equation 6.17:

$$\theta_{\delta}(t) = 2D_p(\theta_s - \theta_m(t)), \tag{6.19}$$

where θ_s is the desired location of the max-div point, $\theta_m(t)$ is the current estimated location of the max-div point, and D_p is a proportional gain. All control gains used were tuned empirically over a set of pre-experiment trials and used for all angles of approach. The same time-to-contact reference value was used for all trials of the max-div control scheme.

6.5.3 Implementation of graze-landing model

Estimates of time-to-contact were obtained using a single 2×10 vector horizontal strip centred on the image centre. Motion of the ground plane was measured by taking the average of the horizontal components of optical flow vectors within the patch, and from this, time-to-contact was calculated using the equation proposed by Srinivasan *et al.* [2000]:

$$\tau = \frac{1}{\tan(\theta) f}, \tag{6.20}$$

where θ is the angle of descent, and f is the translational image velocity. By keeping f constant during the descent, Srinivasan *et al.* proposed the following model of velocity reduction:

$$\begin{aligned} T_x(t) &= f Z_o e^{-fB(t-t_o)}, \\ T_z(t) &= B T_x(t), \end{aligned} \tag{6.21}$$

where Z_o is the initial height above the ground at time t_o, and $B = tan(\theta)$ (*i.e.,* the ratio of T_z to T_x). Figure 6.17 shows example curves of this model for approach angles (*i.e.,* $90^o - \theta$) of 67.5^o and 45^o. In implementing this model for closed-loop control, Equation 6.20 was fed into the same PD control scheme as applied in max-div on-board experiments.

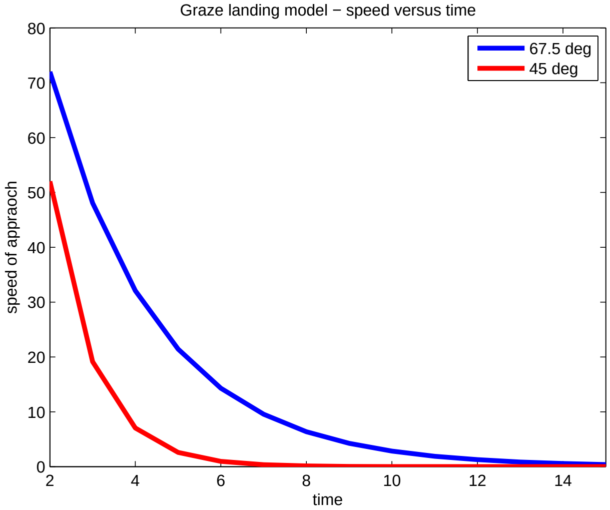

Figure 6.17: Sample curves showing theoretically predicted velocity decay (Equation 6.21) during graze landing approaches, as proposed by Srinivasan *et al.* [2000].

Unlike the max-div control scheme, the graze-landing model requires explicit knowledge of the angle of approach to the surface. As a result, it was not possible to apply the same control parameters across all trials. To maximise the model's ability over the range of approach angles examined, it was necessary to tune the control parameters for each angle of approach. While for grazing approaches, this required only an adjustment of the τ_{ref}, more frontal approaches also required scaling down of the proportional and derivative gains.

6.5.4 Results

6.5.4.1 Closed-loop simulation results

Figure 6.18 shows velocity profiles for each of the simulation trials conducted. Each profile shows recorded velocities from each trial, along with the theoretically expected velocity curve. From these results it can be seen that the recorded velocity profiles exhibit the same exponential behaviour as proposed by Srinivasan *et al.* [2000] for landing honeybees (shown in Figure 6.17). Note that the noise amplitude was kept constant throughout each trial, and thus has greater effect as the velocity decreases.

Figure 6.19 shows sample trajectories of approach for each simulation trial. Results

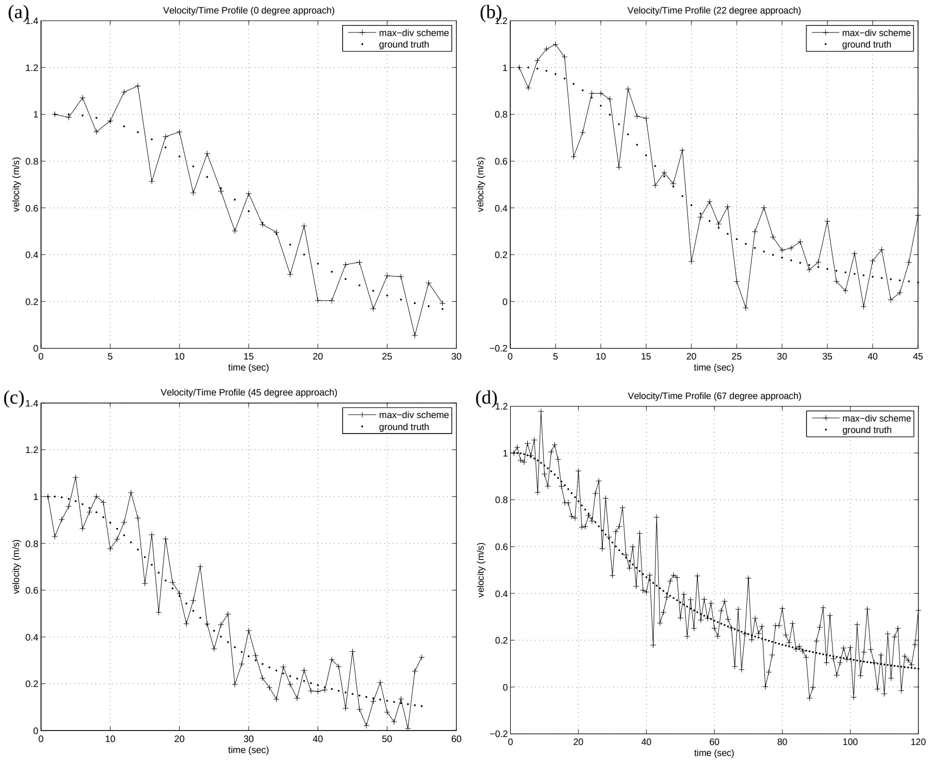

Figure 6.18: Velocity-time profiles for each sample angle of approach: (a) 0^o approach, (b) 22^o approach, (c) 45^o approach, and (d) 67^o approach. In each sequence, Gaussian noise with standard deviation 0.2 was added to the output velocity of the control.

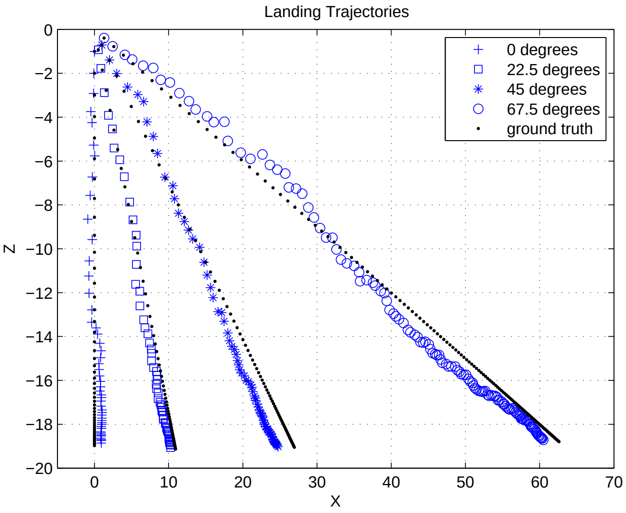

Figure 6.19: Trajectory profiles for each sample angles of approach. In each sequence, Gaussian noise with standard deviation 0.2 was added to the vehicle velocity, and its direction of heading at each time step.

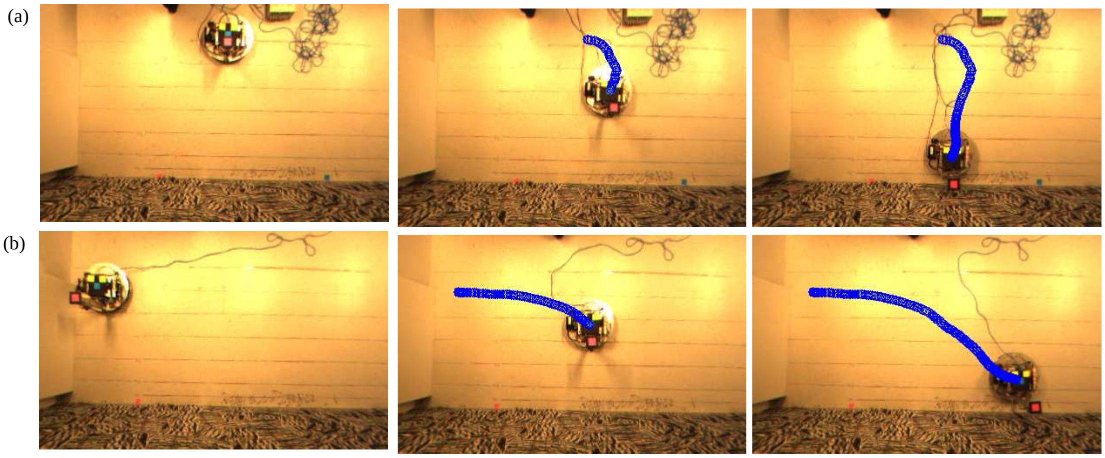

Figure 6.20: Sample overhead images from closed-loop onboard experiments: (a)0^o trials, (b) 67.5^o trials. Blue line shows the plotted path of the robot from overhead tracking.

for all angles of approach show the control law successfully maintaining an average approach angle close to the intended angle.

6.5.4.2 Mobile robot landing/docking results

Across all angles of approach, the max-div control scheme was observed to consistently achieve a safe and stable approach towards the surface, and final stopping distances in close proximity with the surface. Figure 6.20 shows sample overhead images taken during trials of the each discrete angle of approach examined. Video footage from trials of the 0^o and 67.5^o approaches are provided on the thesis CD-ROM. Videos 6.3a and 6.4a show images from the onboard camera, along with the max-div point (white cross) and reference max-div location (blue cross), and the divergence image for both trials. Sample images from both videos are provided in Figures 6.21 and 6.22 respectively. Videos 6.3b and 6.4b show footage of the robot performing both trials.

Figure 6.23 shows the plotted paths taken by the robot in each trial. Position coordinates refer to the image location of the centre of the robot in rectified images taken from the overhead camera. Tracking results clearly distinguish each desired angle of approach, and in general, exhibit consistency across each trial set.

Table 6.4 shows the average angle of approach and stopping distances recorded (measured from the docking surface) for each trial. Stopping distances grow marginally

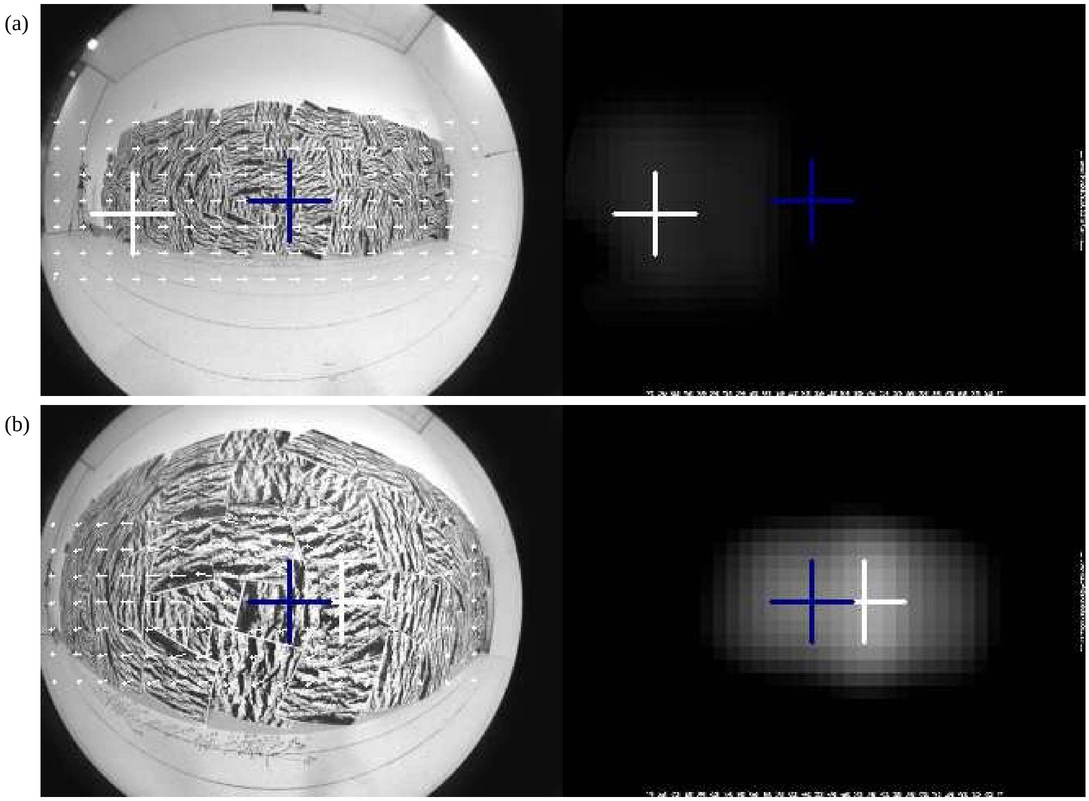

Figure 6.21: Sample output images taken from the 0^o max-div trials for (a), the beginning of the trial, and (b) towards the end. Left image shows the view from the onboard camera, and the optical flow used to compute divergence. The right image shows the divergence image, where greater intensity represents increasing divergence. The blue cross in both images represents the reference max-div location. The white cross shows the current estimated location of the max-div point. For complete footage of trial from the onboard camera, see Video 6.3a on the thesis CD-ROM.

Trial	Mean approach angle	Mean stop distance
0^o	-4.8^o	11.7 cm
22.5^o	23.4^o	12.4 cm
45^o	30.3^o	14.0 cm
67.5^o	53.3^o	14.6 cm

Table 6.4: Performance statistics for closed-loop, on-board trials of max-div control scheme

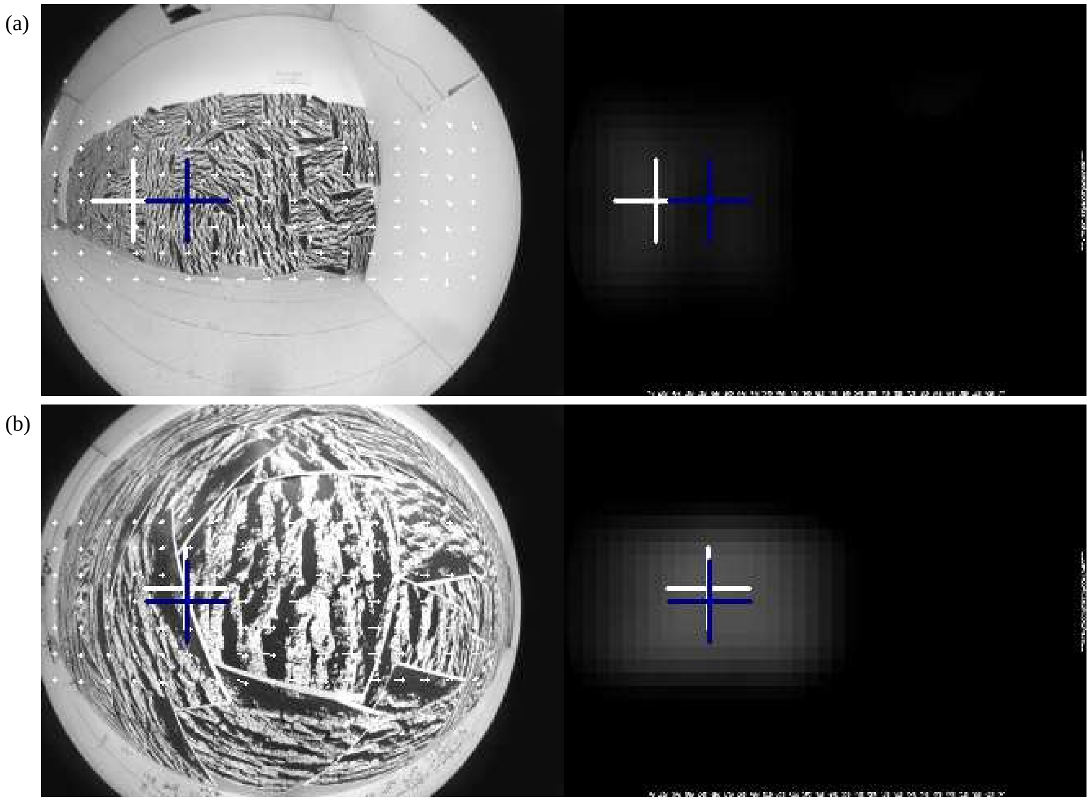

Figure 6.22: Sample output images taken from the 67.5^o max-div trials for (a), the beginning of the trial, and (b), towards the end. Left image shows the view from the onboard camera, and the optical flow used to compute divergence. The right image shows the divergence image, where greater intensity represents increasing divergence. The blue cross in both images represents the reference max-div location. The white cross shows the current estimated location of the max-div point. For complete footage of trial from the onboard camera, see Video 6.3a on the thesis CD-ROM.

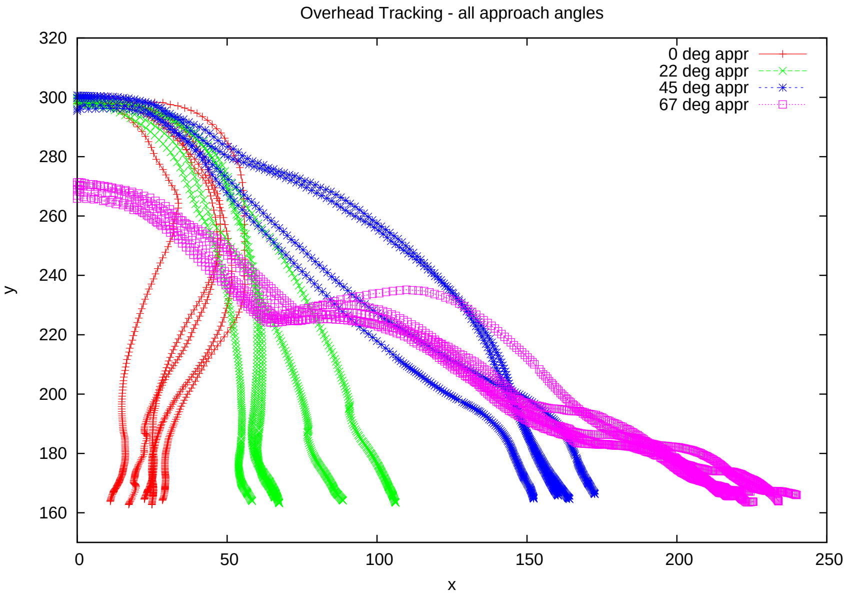

Figure 6.23: Plotted paths taken from overhead tracking of robot for each trial using max-div approach angle control.

with increasing approach angles, but are contained to within a 3cm range. Variation in the approach angle, and final stopping location, is more apparent for near frontal approaches (0^o and 22.5^o). Table 6.4, however, indicates greater accuracy in achieving the desired angle of approach for near frontal approaches.

To examine the distribution of max-div locations during each trial set, Figure 6.24 provides histograms of the relative frequency of plotted max-div locations in the image recorded over each set of trials of the max-div. Across all approach angles, the control scheme appears to successfully maintain maximum divergence about the reference location. Some widening of the distribution is apparent for the near frontal approach, however, it should be noted that the required heading adjustment was greatest for these trials given the initial direction of motion.

Figure 6.25 shows velocity-time profiles obtained from closed-loop onboard trials of the max-div control scheme, and for comparison, the graze-landing model. Velocity curves obtained from the max-div trials, in general, resemble the exponential decay proposed by the graze-landing model (over grazing approaches). During the max-div trials, velocity often begins to fluctuate in the final stages of docking. At this point,

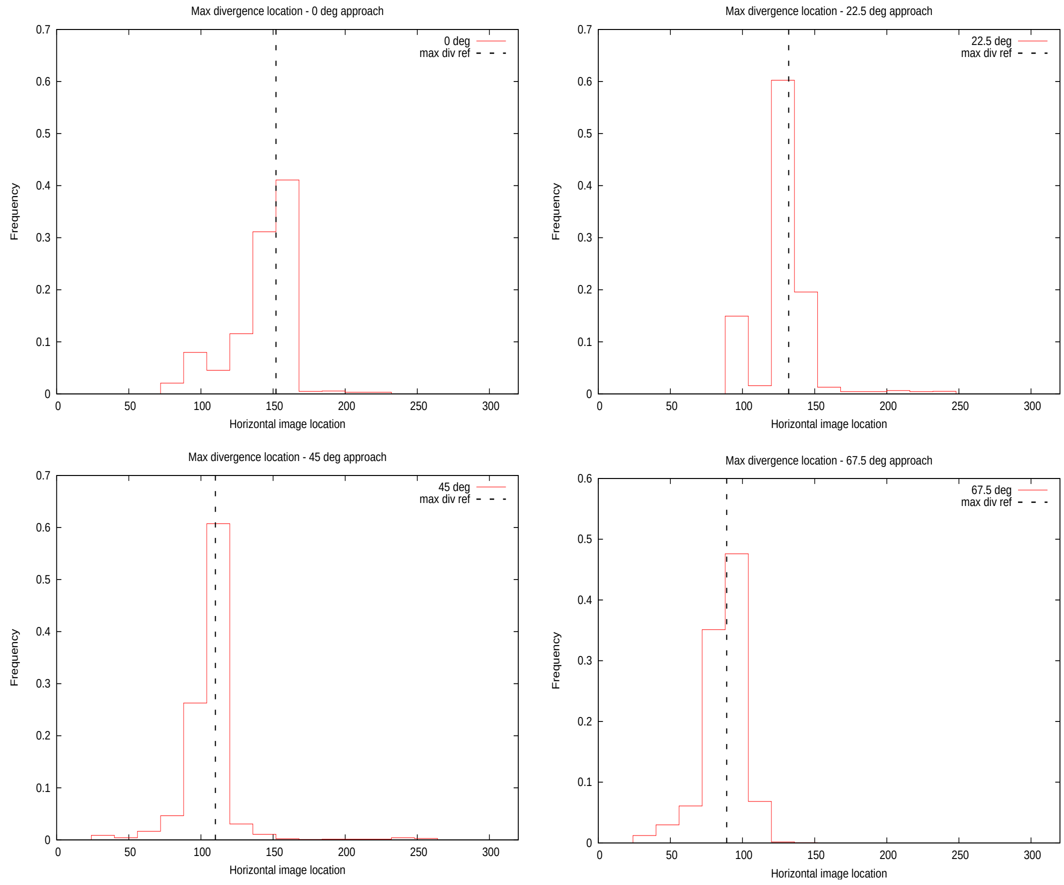

Figure 6.24: Histograms showing relative frequency of max-div image locations recorded during closed-loop onboard trials. Dotted line shows the reference max-div location used for direction control in each trial set.

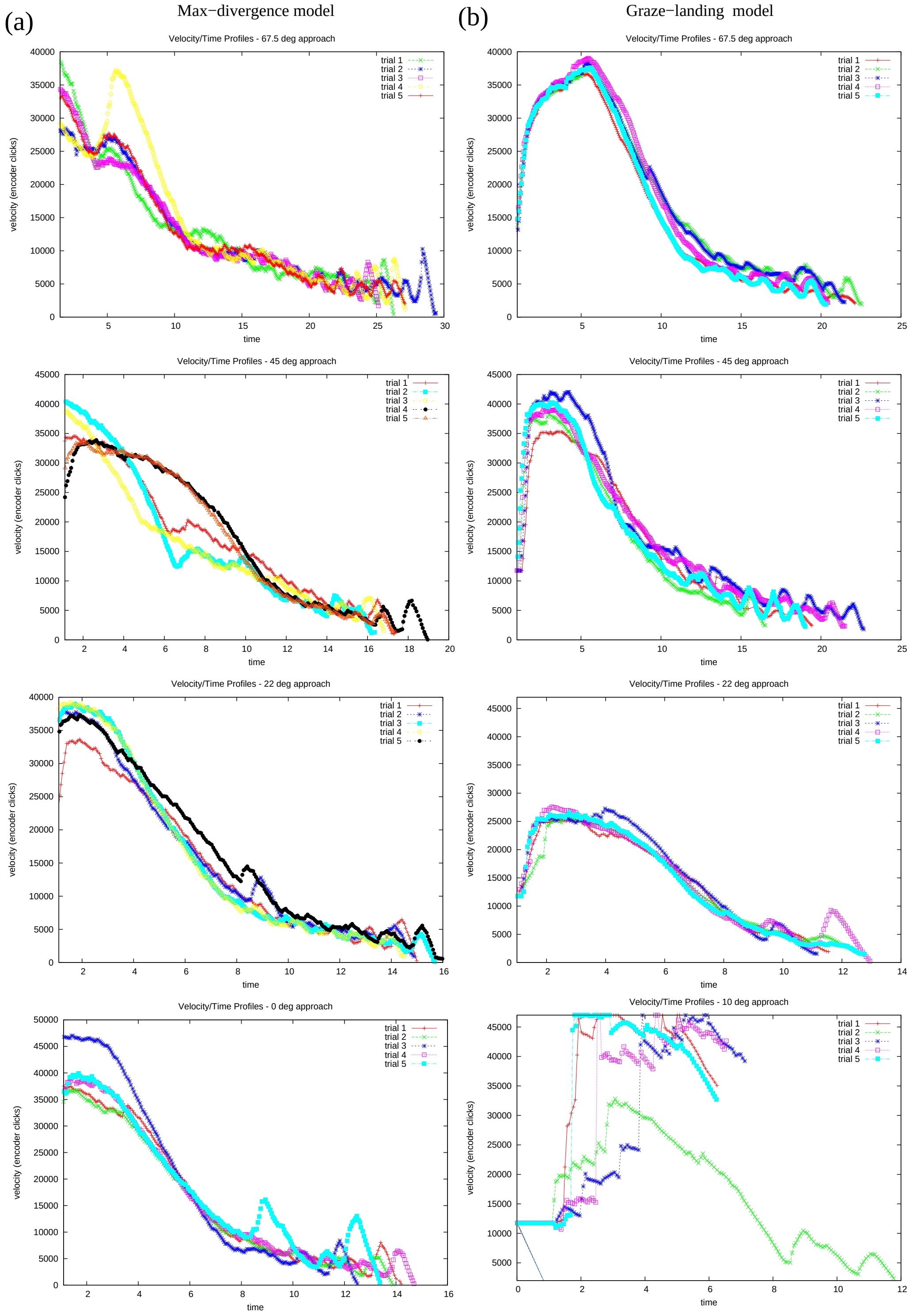

Figure 6.25: Velocity-time profiles recorded during all on-board trials for: (a) the *max-div* model, (b) the *graze-landing* model. Note that no meaningful results were obtained at 0^o using the graze-landing model. Instead we show results for a 10^0 approach, the observed fail case of the graze-landing model.

the robot often performed a series of small thrusts towards the wall before coming to a final halt.

Similar effects are also apparent in graze-landing model trial results. As expected, the graze-landing model provides high stability and consistency for shallow approach angles. Successful trials of the graze-landing model were recorded for approach angles greater than 10^0 (12.5^o being the last trial set for which five successful trials were recorded). However, trial performances were generally observed to degrade from 22.5^o. From this point, control became increasingly sensitive to parameter tuning in order to achieve successful docking performances. At 22.5^o, for example, Figure 6.25 shows velocity profiles with a significantly reduced maximum velocity, a direct consequence of the reduced proportional gain. At 10^o, no control parameters were found to achieve a consistent performance using the graze-landing model. Thus, 10^o was deemed the model's fail case.

Figure 6.26 shows time-to-contact estimates recorded during trials of both models. To facilitate proper comparison of time-to-contact estimates, scaling is applied to time-to-contact estimates obtained during graze-landing model trials to account for the different time-to-contact set-points applied. The time-to-contact set-point value is successfully maintained across all trials of the max-div control scheme, and time-to-contact estimates appear to reflect the consistency of performance exhibited in the velocity profiles. Time-to-contact results for the graze-landing model are also consistent with velocity profile results. A clear loss of stability is apparent for the 10^o fail case of the graze-landing model.

6.6 Discussion

6.6.1 Generality of max-div docking scheme

Open and closed-loop results provide clear evidence of the proposed max-div docking scheme's ability to perform controlled approaches to surfaces of arbitrary orientation. In particular, the consistency of velocity curves and stopping distances provide strong validation of the generality of the proposed landing model.

It is important to note that the max-div control scheme required no alteration to

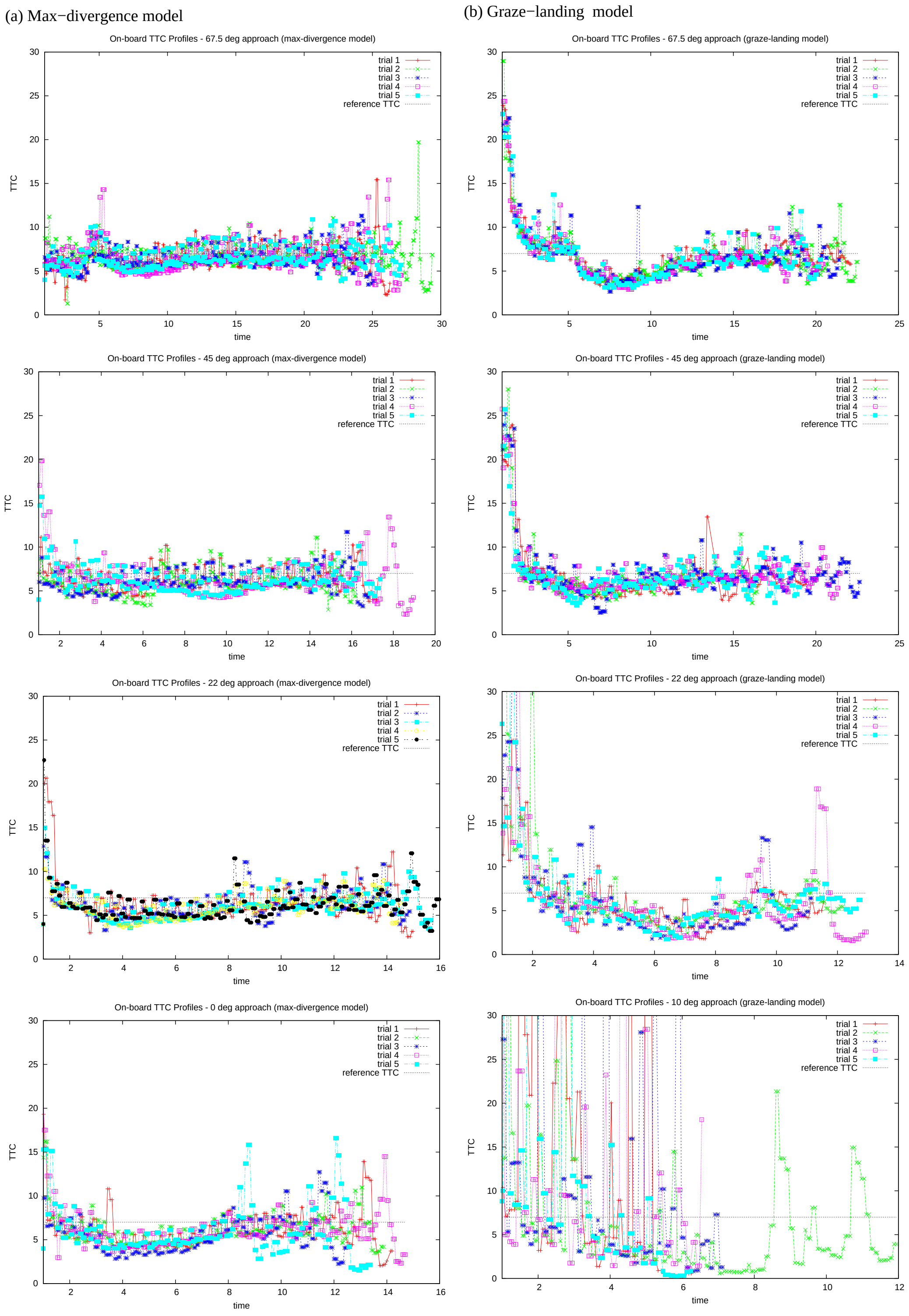

Figure 6.26: Time-to-contact values recorded during all on-board trials for: (a) the *max-div* model, (b) the *graze-landing* model. Note that no meaningful results could be achieved at 0^o using the graze-landing model. Instead we show results for a 10^0 approach, the observed fail case of the graze-landing model.

control parameters to achieve these results. This is in contrast to the graze-landing model which required tuning adjustments for each discrete angle of approach examined. Sensitivity to this control tuning increased as the angle of approach became near-frontal. In contrast, no significant degradation in performance was observed in the max-div control scheme over the range of approach angles examined.

Across all angles of approach, the max-div control scheme provides reasonable stability in maintaining the time-to-contact set-point. In contrast, instability becomes increasingly evident for the graze-landing model as the surface alignment becomes frontal, eventually failing at 10^o. This instability is reflected in velocity curves shown in Figure 6.25(b) for 12.5^o trials. Step-wise fluctuations in velocity, and general inconsistency in the profiles appear to be the result of noisy time-to-contact estimates.

6.6.2 Robustness of time-to-contact estimation

In open-loop experiments, robust time-to-contact estimates were achieved from spherical flow divergence across a wide range of approach angles. While some loss of accuracy was observed in time-to-contact estimates for shallow approaches (*e.g.,* 67.5^o), this increased error appears to be well contained and within workable levels. Notably, on-board 45^o and 67.5^o trials indicate no significant impedance in performance compared with the more frontal approach angles. Improved tuning of the control should reduce fluctuations observed in the final stages of docking for each trial set.

The ability of the max-div scheme to operate in noisy, real-world conditions is well supported by results obtained from outdoor open-loop experiments. These results show that time-to-contact estimates remain workable under both varying lighting conditions, and where surface texture is minimal. While some loss of temporal consistency in time-to-contact estimates is apparent, it is important to note that both sequences depict the motion of a hand held camera, and thus are subject to significant variation in their trajectory. Notably, no loss of accuracy was apparent when rotation was introduced.

6.6.3 Robustness of velocity control

Velocity curves shown in Figure 6.25(a) for the max-div control scheme indicate high stability and consistency in performance was achieved across all trials. The robustness of the scheme is further supported by overhead tracking plots in Figure 6.20, showing consistent performances despite significant variation in approach angle during each trial. Final stopping distances from the wall were also highly consistent. No collisions were recorded in any trials of the max-div control scheme.

These results highlight the invariance of the max-div control scheme to the angle of approach. While better stability in velocity control is evident from the graze-landing model for non-frontal approach angles, this is assisted by the explicit tuning of control parameters for each angle of approach examined. In contrast, the max-div control scheme requires no such tuning adjustments to achieve stable velocity control for each angle of approach. The removal of any explicit representation of approach angle in the max-div control scheme appears to reduces control tuning demands, allowing its more general application. This is in contrast to both the graze-landing model, and the FOE-based docking control scheme presented in Chapter 5, both of which require knowledge of the surface alignment when applied to velocity control.

6.6.4 Robustness of heading control (approach angle regulation)

The robustness of heading control is supported by the underlying stability of the max-div point, as shown in histograms presented in Figure 6.24. The max-div location in the image was successfully maintained across each trial set. Quantitatively, errors between the set and maintained approach angles increase as the surface orientation with respect to the camera moves away from fronto-parallel. Such errors are a likely result of initial alignment errors. The imprecise conditions of these experiments prevented any means of ensuring alignment was accurate. During egomotion, any initial alignment of the camera and surface was subject to variation as a result of unintended robot rotations. Such errors accumulate over time, therefore having greatest influence on trials involving longer travel distances (*i.e.,* 45^o and 67.5^o trials).

In examining quantitative results, it should also be noted that the angular resolution used in on-board trials ranged from 7.2^o about the image centre, to 8.5^o in the periphery. This places a lower bound on the error of the estimated angle of approach. All results shown in Table 6.4 fall within their respective error bounds.

The emphasis on heading control is to maintain a constant angle of approach, rather than a specific angle. In this regard, the control scheme successfully achieves highly robust, and consistent trajectories towards the surface.

6.6.5 Max-div model limitations and issues

6.6.5.1 Existence of a global divergence maximum

The max-div model assumes the existence of the global divergence maximum in the image. Where only a restricted field of view exists, or where the finite landing surface does not occupy the full image area, the global maximum may not lie within the image. Assuming some surface patch projecting onto a portion of the arc connecting $\hat{t}$ and $\hat{n}$ (if the surface were to extend this far), then the local maximum will occur at a point closest to the location of the global divergence maximum. Thus, maintaining constant divergence at the local maximum within the projected area of the looming surface should still provide the best input for velocity control. Assuming continuous motion towards the surface, the local maximum will continue to move towards the global maximum, and thus become more accurate as proximity with the surface increases. However, for such a scenario, use of the max-div point to regulate heading direction would not be appropriate as the divergence maximum will not be stable. This constraint highlights the advantages of employing a full view sphere for navigation tasks.

6.6.5.2 Divergence estimation

The overall stability of the max-div scheme suggests the underlying divergence estimation was consistent and accurate enough to support the underlying control. This was achieved with minimal post-filtering. Unsurprisingly, divergence estimates were noisiest when measured during shallow approaches, at large distances. However, results from all experiments show that the reliability and consistency of divergence estimates improves as the surface nears. This suggests the control itself can be used to maintain visual conditions suitable for obtaining robust divergence estimates. It was also observed that large spatial support for optical flow estimation, and subsequent divergence estimation improved the consistency of results significantly. This was particularly apparent in both outdoor environments examined in open-loop trials. Low resolution divergence estimates computed over large overlapping patches were found to provide the most stable time-to-contact estimates. This was observed to provide robustness to reduced texture, surface inconsistencies, and local flow field errors.

The results presented provide compelling support for the viability of a divergence-based time-to-contact estimation strategy for performing controlled approaches to planar surfaces of any orientation. The proposed model unifies previous approaches to the problem and eliminates the need for any distinction between near fronto-parallel and grazing approaches.

6.7 Implications for visuo-motor navigation

Based on the contributions presented in both this chapter and the previous, we briefly compare both approaches and discuss broader outcomes for the use of optical flow under a visuo-motor framework.

6.7.1 Global flow field invariants

The docking/landing schemes presented suggest robust, simple and efficient visuo-motor control schemes can be developed if frame-to-frame system dynamics are handled in the image domain. This can be achieved by computing visual control inputs, such as time-to-contact, with respect to global invariants of the flow field. (*e.g.,* the FOE, max-div, *etc.*). This design feature is central to the demonstrated robustness of both docking schemes presented in this thesis. In the context of time-to-contact estimation, the use of global invariants improves overall system performance in two ways:

1. the shifting location of global invariants directly reflects frame-to-frame changes in camera motion with respect to the looming surface. Thus, noisy onboard conditions are largely handled in the image domain, reducing the need for complex control schemes.

2. global invariants provide stable locations in the flow field from which to estimate time-to-contact. Both the FOE (in the fronto-parallel case) and max-div points mark locations where divergence is maximal, and local flow estimates are well defined. Both locations therefore provide the most accurate locations in which to estimate time-to-contact.

Notably, these improvements relate to different aspects of the control design, and thus are likely to have a compounding effect on overall system performance. This appears evident in the highly robust performances achieved using relatively simple control schemes, and with minimal tuning requirements.

6.7.2 FOE versus max-div for docking and landing

While the proposed FOE-based estimation scheme presented in Chapter 5 relaxes limitations on time-to-contact estimation and surface alignment, the scheme is still limited to fronto-parallel approaches, and small frame-to-frame rotations.

Such limitations are not present using the max-div control scheme and a spherical projection model. In contrast to perspective projection, spherical projection allows the max-div scheme to circumvent issues of misalignment between the optical and motion axes present under perspective projection. The result is a more general and robust solution to the docking problem.

6.7.3 Visual input choices

As discussed in Section 4.4, other cues such as flow magnitude and closed-contour area provide alternative cues for estimating surface proximity. The implied existence of looming surfaces during the approach, however, provides ideal conditions for the estimation of divergence directly from the flow field. Moreover, these conditions continue to improve during the approach, as looming increases. The result of this is a control scheme that is most stable when most needed (*i.e.,* when impact is imminent).

Notably, the control scheme itself also has a direct influence on the quality of its own visual control inputs. Controlling motion to maintain a sufficiently high divergence level, for example, ensures visual conditions remain well suited for subsequent divergence estimates during the approach. Integrating this bi-directional relationship between control and input through appropriate visual cue choices reduces the reliance of the system on the quantitative accuracy of algorithms used for visual cue extraction.

6.8 Summary

In this chapter we have presented a visuo-motor control scheme for docking and landing with planar surfaces of arbitrary orientation. This solution is based on two analytically proven properties of the divergence on a view sphere approaching a planar surface:

1. the magnitude of the divergence maximum provides a rotation and surface orientation invariant cue of surface proximity; and,

2. the location of the divergence maximum on the view sphere encodes the camera's angle of approach with respect to a planar surface.

By combining these properties, we have provided a general solution to the docking/landing problem. Simulation and a comprehensive set of real image sequences (indoor and outdoor), have demonstrated the viability of the approach for closed-loop use, and for providing robust estimates of time-to-contact under real-world conditions.

Based on the presented results, we have highlighted a number of broad implications for contact estimation in rapidly changing visual conditions. In particular, we have argued that a wide field of view, spherical projection simplifies the extraction of visual quantities like time-to-contact by removing the constraint of alignment with an optical axis. In addition, we have demonstrated the importance of exploiting global invariants of the flow field when computing visual quantities such as time-to-contact. These features are central to the success of the unified landing model presented in this chapter.

In the next chapter we examine visual contact estimation under a structure-from-motion framework. We examine the use of an insect-inspired global spherical view to support real-time depth map recovery from dense optical flow. We also contrast this approach to contact estimation with the visuo-motor based approaches proposed thus far.

Chapter 7

Real Time Biologically-Inspired Depth Maps from Spherical Flow

7.1 Introduction

In the previous two chapters we proposed novel strategies for achieving robust visuomotor control directly from time-to-contact. In this, we considered visual contact estimation without the requirement for full structure and egomotion recovery. However, perceiving scene structure across the visual field is often a prerequisite to higher level navigation tasks such as mode selection, or the coordination of lower-level visuomotor behaviours. This motivates further consideration of how structure-from-motion techniques may best serve the needs of navigation and visual contact estimation.

In Section 3.7 we discussed a number of issues impeding the application of full structure-from-motion algorithms to real-time navigation tasks. Addressing these issues, researchers have considered the use of an insect-inspired spherical projection model over a global view of the scene. In particular, we noted Nelson and Aloimonos [1988], who derive a potentially real-time egomotion estimation algorithm on the view sphere. Currently, however, there exists no thorough examination of the algorithm's performance under real-time constraints, or under real-world conditions. While such techniques offer potential support for real-time structure-from-motion recovery, the rapid recovery of dense depth maps across the visual field is yet to be demonstrated.

In this chapter we propose a scheme for obtaining real-time 3D relative depth maps from a spherical sensor. We present experimental results examining the accuracy and robustness of the strategy, and discuss its potential application to real-time autonomous navigation. In addition, we assess the performance of the Nelson and Aloimonos egomotion estimation algorithm in terms of accuracy, and as a basis for recovering dense 3D relative depth maps in real-time.

The chapter is structured as follows. Section 7.2 presents the proposed depth map recovery scheme, incorporating the egomotion estimation algorithm proposed in [Nelson and Aloimonos 1988]. Section 7.3 presents all experiments conducted to assess performance of the scheme, and results obtained. A discussion of these results is then given in Section 7.4. Section 7.5 provides a discussion of implications for flow-based navigation in general, based on the work presented in this chapter, and previous chapters. A chapter summary is provided in Section 7.6.

7.2 Estimating depth maps from spherical flow

To compute a 3D depth map, we first consider the estimation of egomotion parameters from optical flow on the full view sphere. The Nelson and Aloimonos algorithm proposes the use of one dimensional flow in the direction of great circles on a unit view sphere, about each rotation axis. It is therefore necessary to first decompose the flow field into components along three orthogonal great circles. Nelson and Aloimonos do not explicitly show this step, and so we provide it here.

7.2.1 Decomposition of flow about orthogonal great circles

The position of any point $\hat{p}$ on a view sphere is defined in terms of its angular location along each great circle such that:

$$\hat{p} = \left[\theta_x \, \theta_y \, \theta_z\right], \tag{7.1}$$

where θ_x, θ_y and $\theta_z \in [0, 2\pi]$ are angles in the direction of each orthogonal great circle, $\mathbf{E_x}$, $\mathbf{E_y}$ and $\mathbf{E_z}$ as shown in Figure 7.1.

Considering optical flow under this decomposition, we recall the equation for flow

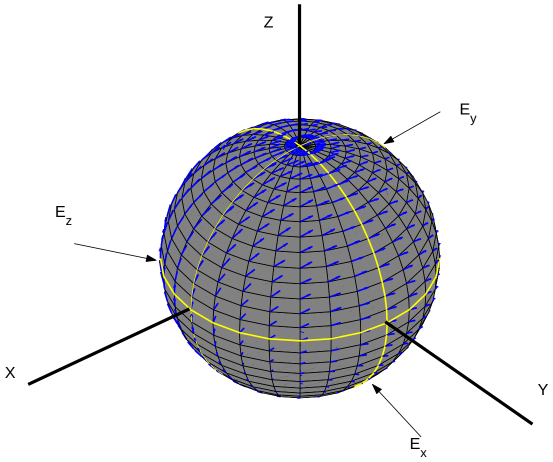

Figure 7.1: Optical flow on the view sphere.

on the unit sphere, $S \in \mathbb{R}^3$:

$$f(\hat{p}) = \frac{v}{|R(\hat{p})|}\Big(\hat{t} - \hat{p}(\hat{t} \cdot \hat{p})\Big) + \hat{p} \times \Omega, \tag{7.2}$$

where $\hat{t}$ is the direction of translation, Ω is the axis of rotation, and $R(\hat{p})$ is the radial distance of the point projecting to $\hat{p}$ on the unit sphere.

We seek to define the 1-dimensional flow in the direction of a great circle on S. Let $\hat{e}$ be the unit vector normal to a plane passing through the centre of S. The intersection of points on S with the plane defined by $\hat{e}$ corresponds to a great circle on S. We define the scalar flow field in the direction of this great circle as as:

$$f_{\mathrm{e}}(\hat{p}) = \hat{q} \cdot f(\hat{p}), \tag{7.3}$$

where

$$\hat{q} = \hat{e} \times \hat{p}. \tag{7.4}$$

The geometric interpretation of this relationship is shown in Figure 7.2.

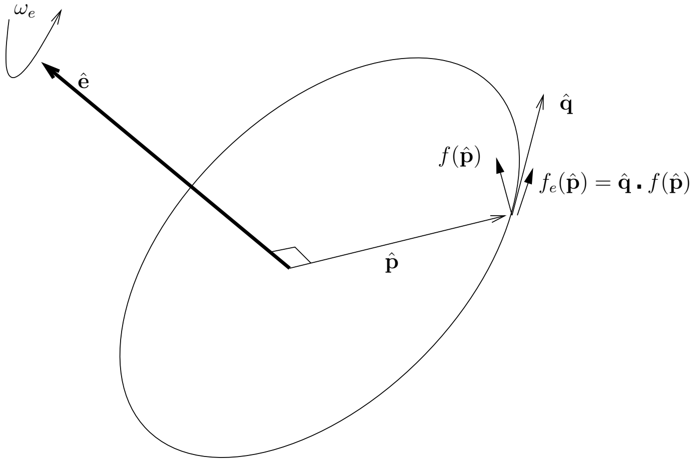

Figure 7.2: Optical flow at a position $\hat{p}$ on the view sphere, projected onto the tangent, $\hat{q}$, of the great circle about the rotation axis $\hat{e}$.

Expanding the right side of Equation 7.3, we obtain:

$$f_{\mathrm{e}}(\hat{p}) = \frac{v}{R(\hat{p})}\Big((\hat{q} \cdot \hat{t}) - (\hat{q} \cdot \hat{p})(\hat{t} \cdot \hat{p})\Big) + \hat{q} \cdot (\hat{p} \times \Omega), \tag{7.5}$$

and noting that $\hat{p}$ is orthogonal to $\hat{q}$, we reduce the equation to:

$$f_{\mathrm{e}}(\hat{p}) = \frac{v(\hat{q} \cdot \hat{t})}{R(\hat{p})} + \omega_{\mathrm{e}}, \tag{7.6}$$

where ω_{e} is the rotational velocity about $\hat{e}$ (*i.e.,* in the direction $\hat{q}$).

The first term in Equation 7.6 defines the translational component of motion. We decompose the projection of $\hat{t}$ onto $\hat{q}$ into two sub-projections. Firstly, a projection of $\hat{t}$ into the plane of the great circle about $\hat{e}$, followed by a projection onto $\hat{q}$, such that:

$$\begin{aligned} v(\hat{q} \cdot \hat{t}) &= v\Big(1 - \cos(\gamma_{\mathrm{et}})\Big)\Big(1 - \cos(\phi_{\mathrm{e}} - \theta_{\mathrm{e}})\Big), \\ &= v \sin(\gamma_{\mathrm{et}}) \sin(\phi_{\mathrm{e}} - \theta_{\mathrm{e}}), \qquad (7.7) \end{aligned}$$

where γ_{et} is the angle between $\hat{e}$ and $\hat{t}$, and θ_e and ϕ_{e} are the angular positions of the projected locations $\hat{p}$ and $\hat{t}$ on the great circle about $\hat{e}$ respectively.

Substituting Equation 7.7 into Equation 7.6, we obtain the scalar flow in the direction of the great circle about $\hat{e}$, such that:

$$f_{\mathrm{e}}(\hat{p}) = \frac{v}{R(\hat{p})}\Big(\sin(\gamma_{\mathrm{et}})\sin(\phi_{\mathrm{e}} - \theta_{\mathrm{e}})\Big) + \omega_{\mathrm{e}}. \tag{7.8}$$

From this representation, we may express any optical flow vector on the sphere in terms of the components of motion in the direction of each orthogonal great circle, such that:

$$f_x(\theta) = \frac{v}{R(\theta)}\Big(\sin(\gamma_{\mathrm{xt}})\sin(\phi_{\mathrm{x}} - \theta_{\mathrm{x}})\Big) + \omega_{\mathrm{x}}, \tag{7.9}$$

$$f_y(\theta) = \frac{v}{R(\theta)}\Big(\sin(\gamma_{\mathrm{yt}})\sin(\phi_{\mathrm{y}} - \theta_{\mathrm{y}})\Big) + \omega_{\mathrm{y}}, \tag{7.10}$$

$$f_z(\theta) = \frac{v}{R(\theta)}\Big(\sin(\gamma_{\mathrm{zt}})\sin(\phi_{\mathrm{z}} - \theta_{\mathrm{z}})\Big) + \omega_{\mathrm{z}}, \tag{7.11}$$

where subscripts indicate the axis about which flow and related parameters are defined.

It is important to note that the above equations are defined for any three great circles lying on orthogonal planes, and are not limited to equators about the X, Y and Z axis. As such, the above equations show that in addition to the translation, flow in the direction of any great circle is effected by only a single rotation about the axis perpendicular to the great circle's plane. This observation has lead to the development of a full egomotion algorithm for optical flow on the sphere.

7.2.2 Egomotion estimation

In Section 3.8.2 we discussed the key geometric properties of optical flow on the sphere exploited by Nelson and Aloimonos to recover full 3D rotational velocities from spherical flow. For convenience, we outline them again here:

1. the component of flow parallel to any great circle is effected only by the rotational component about its perpendicular axis, thus decoupling it from rotations about orthogonal axes.

2. under pure translation, both the FOE and FOC will co-exist at antipodal points on the sphere, and will evenly partition flow along any great circle connecting

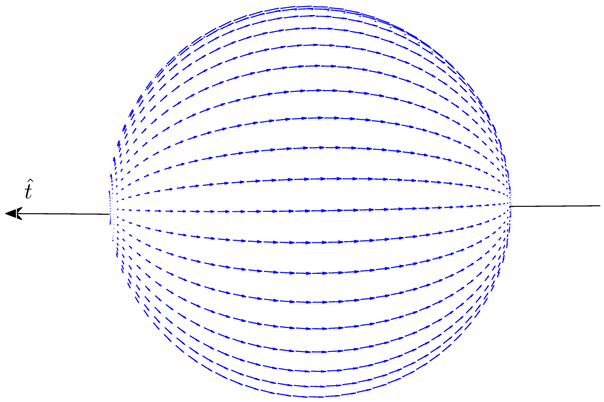

Figure 7.3: After de-rotation (or during pure translation), flow vectors follow great circles passing through the FOE and FOC.

these two point, into two distinct directions of motion (*i.e.,* clockwise and counter-clockwise).

3. the existence of any rotational motion along a great circle causes the FOE and FOC to converge, thus ensuring the two points will only lie at antipodal locations under pure translation.

From these observations, Nelson and Aloimonos propose an algorithm for recovering the rotational component of flow, ω, about any great circle of flow $e(\theta)$. Again for convenience, we reproduce the algorithm in Figure 7.4. By applying this algorithm to great circles about each rotational axis, the complete recovery of the sphere's rotation is achieved. After de-rotation, the direction of translation is also given by the line passing through the FOE and FOC. Figure 7.3 shows an example of this.

The algorithm's run time performance is dictated primarily by the quantisation of discrete locations on the great circle, and the range of possible rotations for each great circle. Given reasonable choices, the algorithm can provide fast execution.

7.2.3 Egomotion estimation under planar motion

Nelson and Aloimonos make use of the full view sphere to estimate egomotion under general motion. We note, however, that if camera motion is constrained to the plane,

```
for $\omega_c = \omega_{min}$ to $\omega_{max}$ do
  $D[\omega_c] = 0$
  for $\theta_c = 0$ to $2\pi$ do
    $a = e(\theta_c) - \omega_c$
    $\beta = \phi - \theta_c$
    if $a < 0$ and $0 \leq \beta < \pi$ then
      $result = -a$
    else if $a > 0$ and $\pi \leq \beta < 2\pi$ then
      $result = a$
    else
      $result = 0$
    end if
    $D[\omega_c] = D[\omega_c] + result$
  end for
end for
$\omega = min\_index(D[\omega_{min} : \omega_{max}])$
return $\omega$
```

[Nelson and Aloimonos 1988]

Figure 7.4: Nelson and Aloimonos egomotion algorithm. In words, for each discrete point, θ_c, on a circle of flow, $e(\theta)$, test a range of rotations by de-rotating flow along the circle. The sum of the residual flow on the circle, after de-rotation, is taken. The chosen rotation is that which yields the smallest sum of flow on the circle after derotation.

rotation and translation parameters may be recovered from any concentric circle of flow on the view sphere centred on the rotation axis (*i.e.,* it need not be the great circle). We show this explicitly below.

Let $\hat{e}$ be the rotational axis, and $\hat{t}$ be the direction of translation within the plane perpendicular to $\hat{e}$ containing the great circle about $\hat{e}$. Let R be the arc on the view hemi-sphere (we need only consider a hemisphere for planar motion) passing through a point $\hat{p}$ on the great circle and $\hat{e}$ (see Figure 7.5). We note that R defines the set of all projected locations of $\hat{p}$ on concentric circles about $\hat{e}$. Let $\hat{r}$ be an arbitrary location on R, representing a concentric circle from which egomotion will be estimated. Substituting $\hat{p}$ for $\hat{r}$ in Equation 7.5, we obtain an equation for flow in the direction of the concentric circle passing through $\hat{r}$:

$$f_{\mathrm{e}}(\hat{r}) = \frac{v}{R(\hat{r})}\Big((\hat{q} \cdot \hat{t}) - (\hat{q} \cdot \hat{r})(\hat{t} \cdot \hat{r})\Big) + \hat{q} \cdot (\hat{r} \times \Omega). \tag{7.12}$$

From this we note two observations:

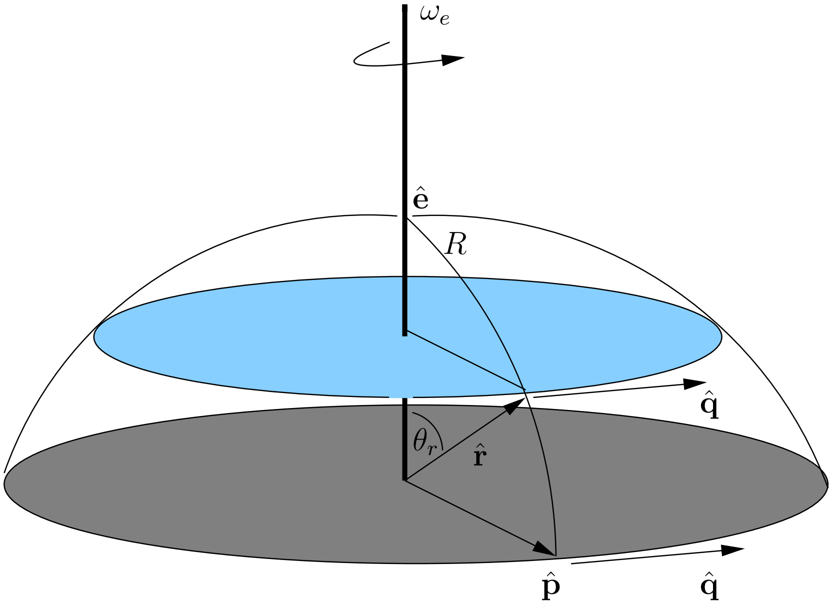

Figure 7.5: Under planar motion, egomotion estimation can be applied to any concentric circle about the axis of rotation. See text for details.

1. $\hat{q}$ is orthogonal to all points along R, and thus $(\hat{q} \cdot \hat{r}) = 0$; and
2. given $\hat{e}$ is the rotational axis, $(\hat{r} \times \Omega) = \omega_e \sin(\theta_r)$, where θ_r is the angle between $\hat{e}$ and $\hat{r}$.

The first observation allows the same reduction of the translational motion term as applied for the great circle case (noting also that $\hat{t}$ is constrained to the plane of the great circle and thus $\sin(\gamma_{\text{et}}) = 1$). The second observation indicates that the rotational velocity along a concentric circle differs from the true rotational velocity by a scale factor $\sin(\theta_r)$. Thus, we reduce Equation 7.12 to:

$$f_{\text{e}}(\theta_e, \hat{r}) = \frac{v}{R(\hat{r})}\Big(\sin(\phi_{\text{e}} - \theta_{\text{e}})\Big) + \omega_e \sin(\theta_r). \tag{7.13}$$

The scale factor $\sin(\theta_r)$ is obtainable from camera calibration. Thus, assuming planar motion, egomotion recovery may be applied equivalently using any concentric circle about the rotation axis perpendicular to the plane. We exploit this property when implementing the egomotion estimation algorithm for all ground-based experiments reported later in this chapter.

7.2.4 Generating relative depth maps

After the removal of rotation components from optical flow on the view sphere, all optical flow vectors follow great circles passing through the FOE and FOC (as shown in Figure 7.3). Thus, we may express the magnitude of the residual translational optical flow at a discrete location θ_{e} on such a great circle as:

$$f_{\mathrm{e}}(\theta_e) = \frac{v}{R(\theta_{\mathrm{e}})}\Big(\sin(\phi_{\mathrm{e}} - \theta_{\mathrm{e}})\Big). \tag{7.14}$$

Note that $\sin(\theta_{\mathrm{e}})$ is now set to 1, as $\hat{t}$ lies in the plane of all great circles passing through the FOE and FOC.

Assuming a static environment, we may define an equation for obtaining the radial distance to any scene point projecting onto a great circle passing through the FOE and FOC as:

$$R(\theta_{\mathrm{e}}) = \frac{v}{f(\theta_{\mathrm{e}})}\Big(\sin(\phi_{\mathrm{e}} - \theta_{\mathrm{e}})\Big). \tag{7.15}$$

Given egomotion recovery provides ϕ, the only remaining unknown is the sensor's translational velocity. In general, however, this is unavailable, and so depth can only be estimated up to a scale factor such that:

$$\frac{R(\theta_{\mathrm{e}})}{v} = \frac{1}{f(\theta_{\mathrm{e}})}\Big(\sin(\phi_{\mathrm{e}} - \theta_{\mathrm{e}})\Big). \tag{7.16}$$

While not an absolute measure, this provides a relative depth map which is sufficient for most navigation tasks.

It is important to note that Equation 7.16 is only defined where optical flow exists (*i.e.,* $f_{\mathrm{e}}(\theta_{\mathrm{e}}) \neq 0$). Thus, range cannot be reliably measured where a lack of texture exists, or where flow magnitude tends to zero such as at the FOE and FOC. Given a spherical view of a sufficiently textured environment, however, enough features should exist away from these singularities to obtain workable depth maps for scene structure recovery.

Table 7.1: Simulation error measures

noise	$\hat{t}$	Ω error			Depth
(std dev)	error	ω_x	ω_y	ω_z	rel error
0^o	3.82^o	1.49^o	1.63^o	1.58^o	3.7%
2^o	3.96^o	1.86^o	2.62^o	3.25^o	6.0%
4^o	8.68^o	1.88^o	2.33^o	1.40^o	11.7%
8^o	10.99^o	1.26^o	1.48^o	1.16^o	17.0%

7.3 Performance evaluation

In this section we examine the performance of the proposed depth map recovery scheme from optical flow under spherical projection. We first present results from simulation testing. We then present a series of experiments examining the performance of the scheme over real image sequences. These experiments provide both quantitative evaluation under controlled conditions, and qualitative assessment in more realistic, real-world scenarios.

7.3.1 Simulation experiment

7.3.1.1 Method and implementation

The proposed depth map recovery scheme was implemented and tested in a Matlab environment. A model of a full unit sphere undergoing general motion in a virtual 3D boxed space was constructed. In each iteration of the simulation, optical flow was computed on the view sphere for varying 3D translation directions and rotations about each principle axis of the view sphere. Random number generation was used to assign values to each motion parameter. Rotational velocities were restricted to the range $[-0.5, 0.5]$ radians per iteration. Translation direction changes were unrestricted. On each equator, 112 discrete points, and 100 possible rotations were used for derotation. Each simulation run consisted of ten iterations. To avoid singularities along the axis of translation, depth map values were not computed for points within a 10^o angular distance about the estimated translational direction (and its antipodal point on the view sphere). To examine robustness, increasing levels of Gaussian noise were added to the angular component of each flow vector estimate for each simulation run.

7.3.1.2 Simulation results

Table 7.1 provides mean errors obtained during each simulation run. Rotational errors are given as the mean of the absolute difference between the estimated and true rotational velocities (given in degrees) for each simulation run. The translational direction error is given as the mean of the angular error between the estimated translational direction and ground truth. Depth map estimation errors are given as the mean of relative errors against ground truth.

Of note in Table 7.1 are the errors obtained for rotational velocities. These remain largely unchanged as noise increases. This is in contrast to other errors, which exhibit a steady increase as noise levels grow larger. These errors, however, remain largely contained. Increases in the translational direction error, for example, appear to diminish as noise grows larger. This is most likely the combined result of the Gaussian noise model applied and the robustness of the egomotion estimation. Depth map relative error appears to grow linearly with noise. This increased error rate is unsurprising given depth estimates will be subject to local flow noise as well as egomotion estimation errors.

While real-world conditions are noisy, for navigation involving predominantly forward motion, translational velocity should remain high with respect to rotation. This should improve both the translation estimates obtained, and the stability of relative depth estimates, subject to flow estimation accuracy.

7.3.2 Real-world experiments

We first provide details of the implementation of the depth map recovery scheme applied in all real-world experiments. We then describe each experiment before presenting the results obtained.

7.3.2.1 Implementation of depth map recovery

The 3D depth map algorithm was implemented for use with a single camera (Unibrain Fire-i BCL 1.2), with fish-eye lens providing a 190^o field of view. The image centre was aligned approximately along the axis of rotation, allowing egomotion estimation to be performed on a circle of flow vectors about the image centre. The fish-eye camera's wide field of view provides a suitable approximation to a hemispherical projection of the scene. Its use of standard CCD hardware provides a simple and relatively cheap alternative to mirror configurations, or more specialised (and expensive) omni-directional camera systems.

Egomotion estimation

Assuming ground-based motion, egomotion recovery was implemented for a single rotation, using a circle of evenly distributed image points (radius 110 pixels) around the estimated projective centre of the camera. Due to limitations imposed by the size of the camera's image plane, the true great circle could not be used, however, as explicitly shown in Section 7.2.3, any concentric circle within the great circle is sufficient for de-rotation. Camera calibration was used to determine the rotation scale factor corresponding to the chosen concentric circle (*i.e.,* $\sin\theta_r$ in Equation 7.13).

Depth map estimation

Depth maps of the environment were generated from the estimated flow field and egomotion parameters. For each image location of interest, relative depth was obtained by first computing its angular displacement from the direction of translation about the image centre (*i.e.,* $\phi_e - \theta_e$, assuming the image centre roughly coincides with the axis of rotation). Depth is then computed as per Equation 7.16. No post-filtering of flow estimates, translational direction or relative depth estimates.

Robot platform

The mobile robot employed for on-board experiments provides an omni-directional drive-system. Motion of the robot was assumed to be constrained to the ground plane, with only a single axis of rotation normal to the ground plane (*i.e.,* the Y axis). The camera was placed centrally on top of the robot facing upward along the rotation axis for all on-board experiments.

7.3.2.2 On-board controlled environment experiment

An artificial workspace of upright planar surfaces was created in a laboratory environment. Surfaces consisted of varying levels of texture. The robot was placed inside the workspace and manually guided through the environment under joystick con-

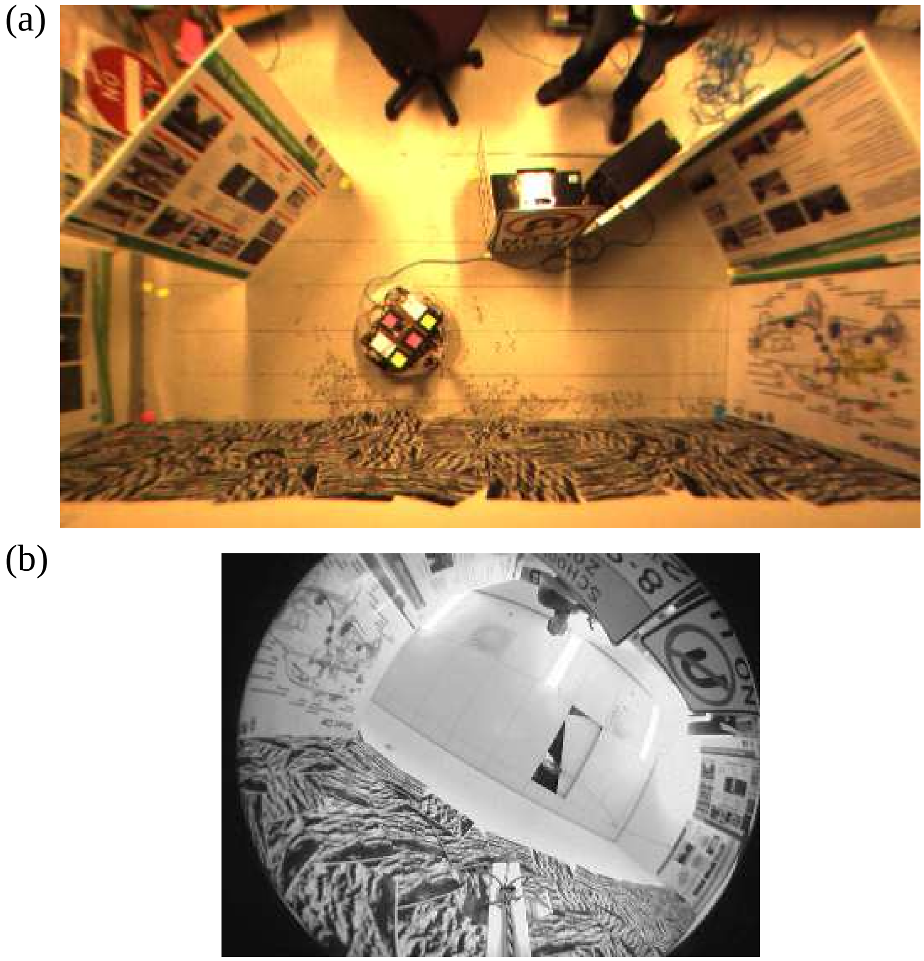

Figure 7.6: Images of workspace for controlled environment experiment showing (a) an overhead view from the camera used for tracking, and (b) an on-board view using the omnidirectional camera.

trol. Figure 7.6(a) shows the robot and the experimental workspace. Images from the on-board camera were captured and used to estimate egomotion, and recover relative depth in the scene. On-board frames were captured at approximately 15Hz, at a resolution of 320 × 240 pixels. Optical flow was computed at full resolution, using a two level pyramidal implementation of Lucas and Kanade's gradient-based method [Lucas and Kanade 1981].

To obtain ground truth data, images of the robot's movement were recorded from a calibrated overhead camera placed approximately four metres above the workspace. Tracking software was used to extract the robot's position, velocity and heading from overhead images. Coloured markers were placed on top of the robot to assist tracking. Tracking data was computed at 6Hz. The image coordinates of all upright surfaces on the ground plane were recorded and used to construct 2D ground truth depth maps with respect to the robots tracked position each frame. Video 7.2 on the thesis CD-ROM provides overhead footage showing the ground truth construction process. Figure 7.6(b) shows a sample overhead image acquired during tracking of the robot as it moves in the workspace. The figure also shows where surfaces in the environment have been marked, and used for ground truth depth map generation.

Figure 7.7 shows the plotted path and orientation of the robot across the sequence. Figure 7.8(a) and (b) show the translational and rotational velocities of the robot as computed from overhead tracking. Note that translation and rotation are subject to significant variation across the sequence, thereby allowing an examination of robustness under varying conditions.

7.3.2.3 Real-world navigation experiments

Two additional image sequences were constructed to examine performance in more realistic environments. In both sequences, the camera's forward velocity was kept approximately constant, while rotation about the axis perpendicular to the ground plane was introduced. Optical flow estimation was run at a single pyramid level only for faster execution. Image resolution was 320 × 240 pixels for both sequences. Frames were captured at approximately 15Hz. To recover 3D structure, depth maps were generated across the image, within a 110 pixel radius of the image centre. We describe

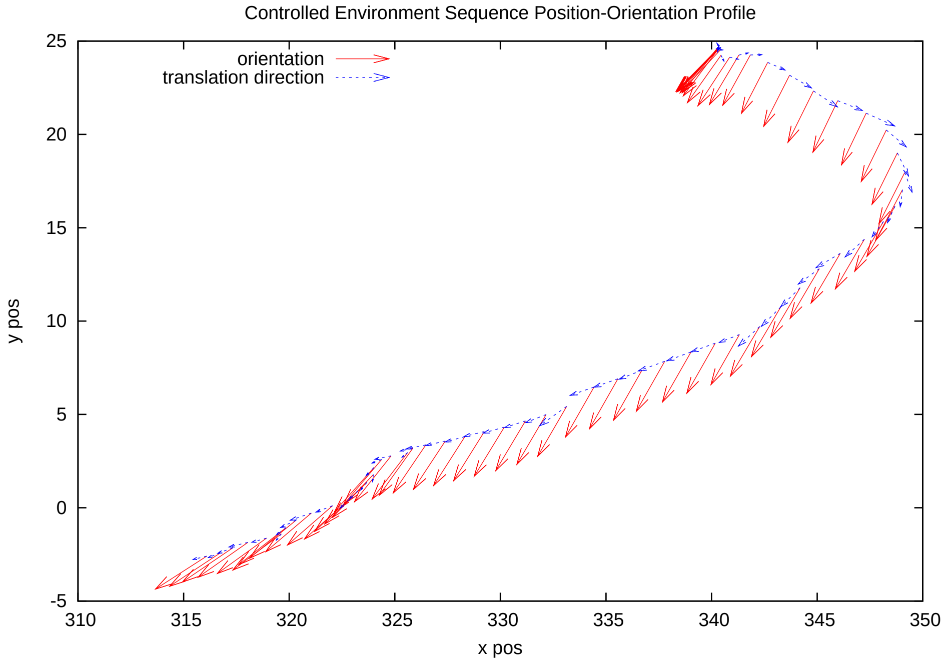

Figure 7.7: The tracked position and orientation of the robot during the controlled environment experiment.

the two sequences below.

On-board corridor navigation sequence: To examine performance in a typical indoor office environment, the robot was manually guided through a series of corridors in our lab. As in the controlled-environment experiment, images were acquired from an on-board camera fixed as close as possible to the robot's rotation axis, facing upwards. Figure 7.9 shows sample images from the sequence.

Hand-held cluttered fly-through sequence: A second sequence was constructed depicting the camera's motion through a cluttered kitchen environment. The camera was hand-held, facing toward the ground plane as it was walked through the kitchen. The motion of the camera is therefore less constrained than in on-board experiments, and the environment less structured. Sample frames from this sequence are given in Figure 7.10.

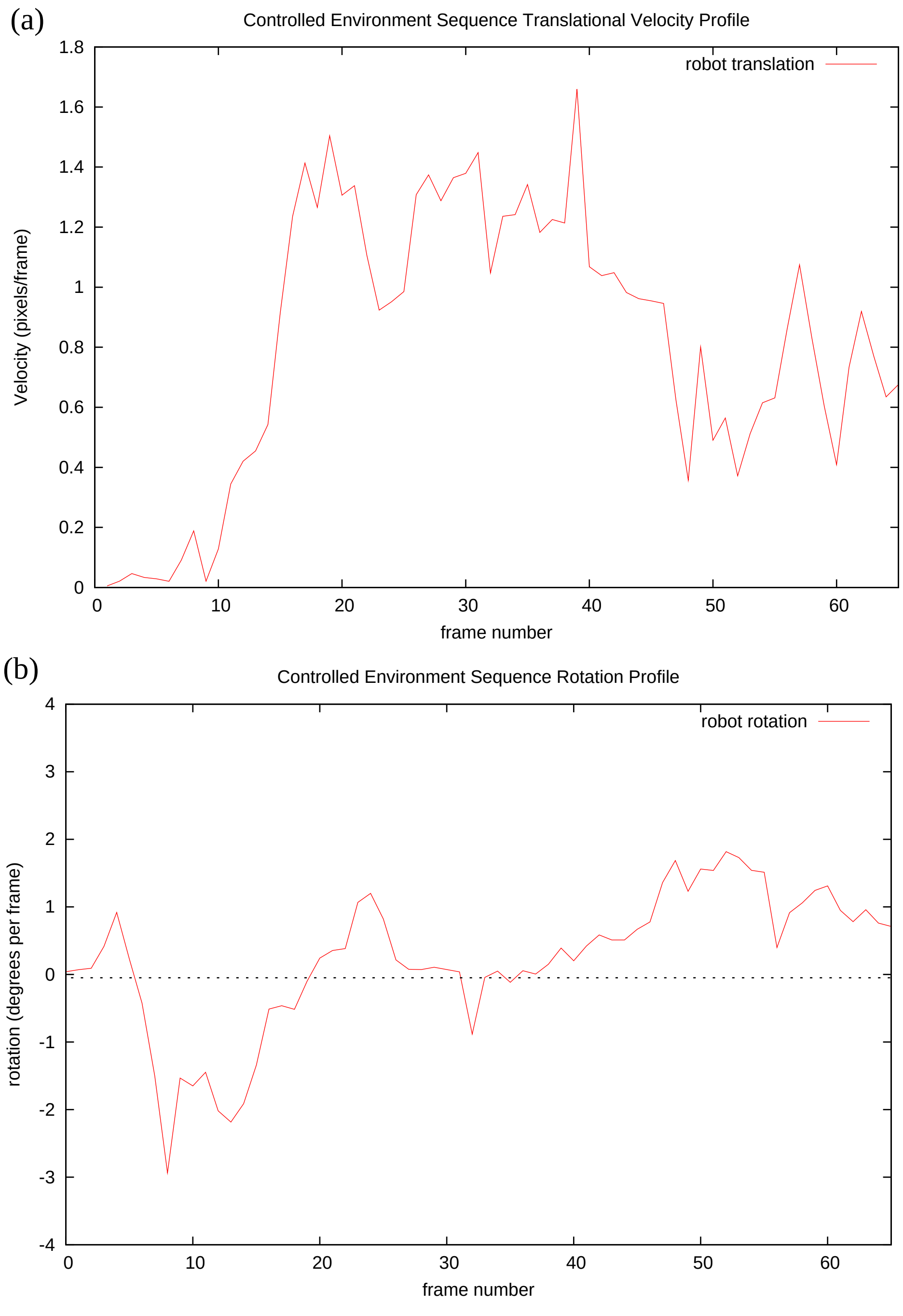

Figure 7.8: Robot motion data recorded during controlled environment experiment, showing: (a) robot translation, and (b) robot rotation. For reference, the dotted line in (b) indicates zero rotation.

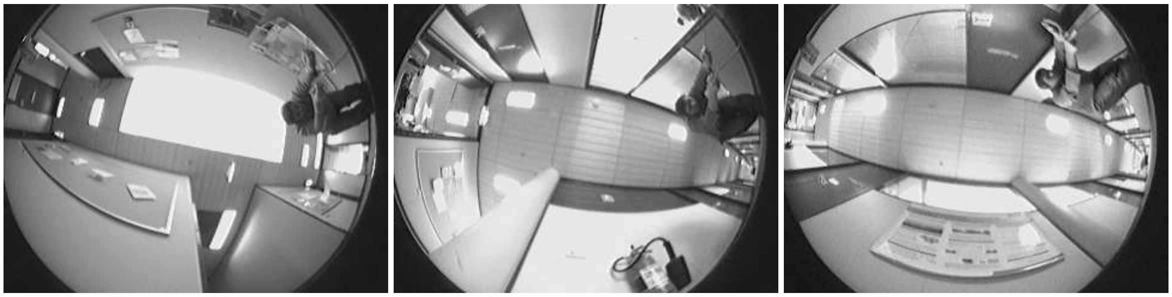

Figure 7.9: Sample frames from the corridor navigation sequence

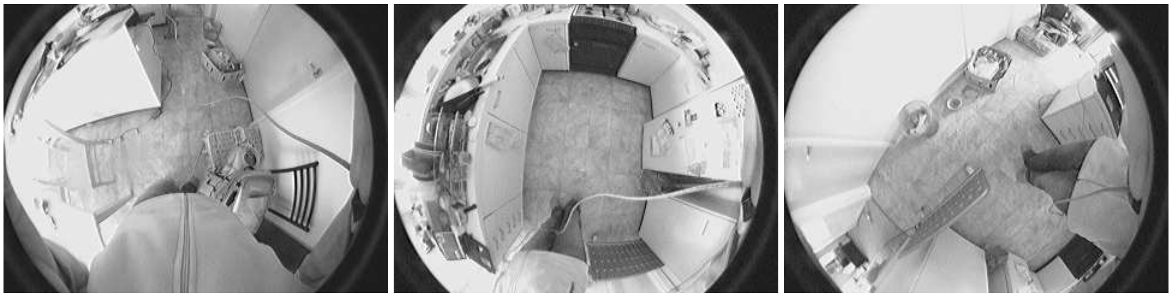

Figure 7.10: Sample frames from the hand-held cluttered fly-through sequence

7.3.3 Results

7.3.3.1 On-board controlled environment results

Egomotion estimation

Estimates of robot translation direction and rotation were recorded for each frame acquired during the robot's motion. This data was compared with corresponding ground truth egomotion data, computed from overhead tracking data at approximately the same time instant. Figures 7.11(a) and 7.11(b) compare estimated translation directions and rotational velocities with overhead tracking data across the sequence.

Translation direction estimation generally follows overhead tracking results closely. While fluctuations are evident in the early frames of Figure 7.11(a), these correspond to a period of low translational velocity (as shown in Figure 7.8(a)) and high rotation (as shown in Figure 7.11(b)) as the robot begins its motion. Rotation estimation provides a similarly close correlation with overhead data. Fluctuations become more significant when rotations grow larger. This is a likely result of small frame-to-frame variations

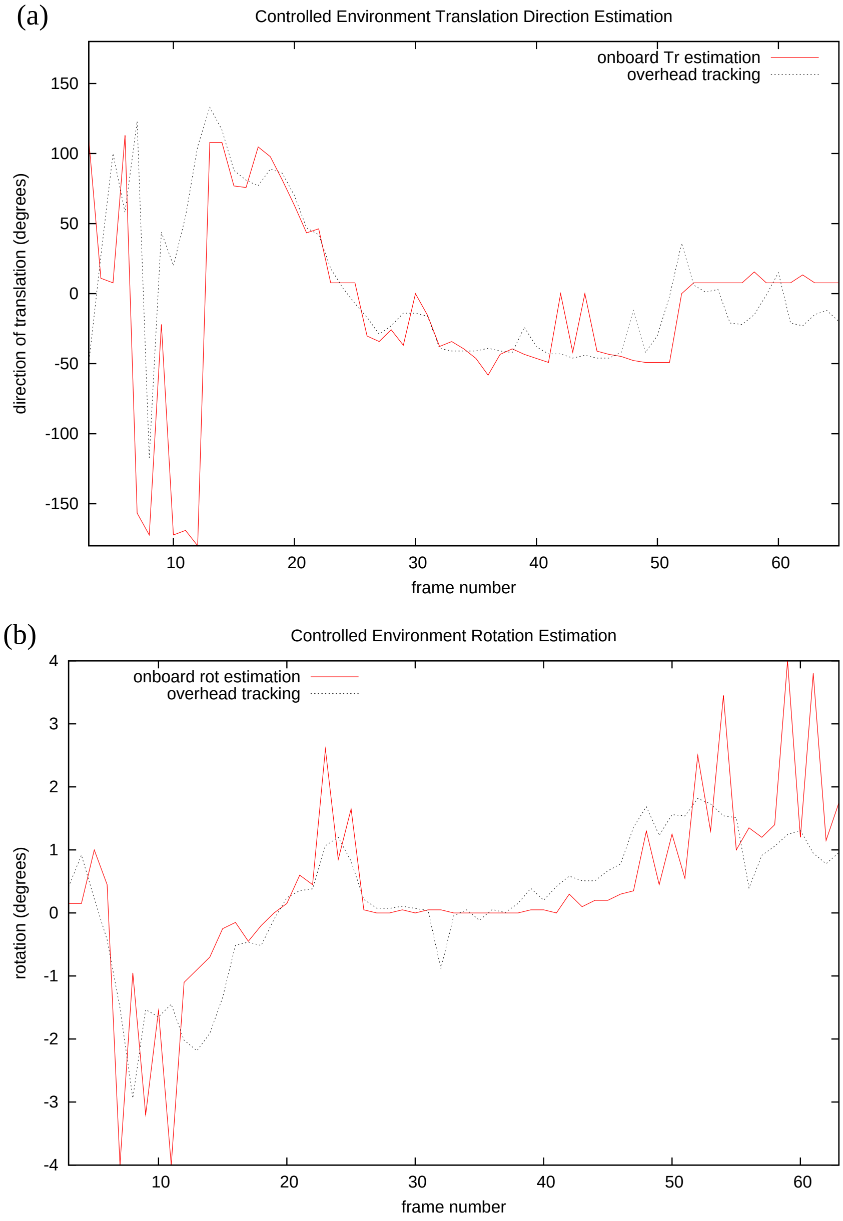

Figure 7.11: Egomotion estimation from controlled environment sequence: (a) camera translation direction, (b) camera rotation.

in on-board frame capture. Such variations over small time intervals are unlikely to be detected in the slower capture rate used for overhead tracking. Occasional dropped frames were also evident in on-board frame capture. The resulting increase in time delay between frames effectively doubles the estimated rotational velocity.

Depth map recovery

Depth maps of the environment were generated from flow vectors in the image periphery, along rays originating from the image centre. For each ray of interest, a single depth estimate was obtained from the average of relative depth estimates measured along the ray, between a radius of 110 and 130 pixels. Depth rays were distributed evenly, 1^o apart. Figure 7.12 shows a sample onboard frame from the sequence.

Figure 7.13 shows sample depth maps estimated across the controlled environment experiment. Samples are given for every ten overhead frames of the sequence. Note that both estimated and ground truth depth maps are scaled by the average depth of their respective maps to allow direct comparison. Figure 7.12 shows the corresponding overhead image, onboard image (with depth map region of interest marked), and a grayscale representation of the estimated depth map (brightness indicating relative closeness of proximity) for each sample. The robot's orientation and estimated translation direction are also shown in both figures. Depth estimates are shown across a 120^o field of view on either side of the axis of translation.

Varying levels of accuracy are evident in Figure 7.13 results. In general, depth estimates for surfaces close to the robot correlate well with ground truth. This is most evident in the top left portion of Figure 7.13(a), and more generally about the right angle surface corners across most of the samples. Accuracy declines rapidly as surface depth increases, as can be seen in the bottom right region of depth maps shown in Figures 7.13(a), (e) and (f).

7.3.3.2 Onboard corridor navigation results

Figure 7.14 shows sample depth maps obtained during the corridor navigation experiment. The first column shows the central image from the buffered frames used to compute the optical flow for the corresponding depth map. The second column shows an intensity map of the relative depths of objects in the scene (brighter is closer) esti-

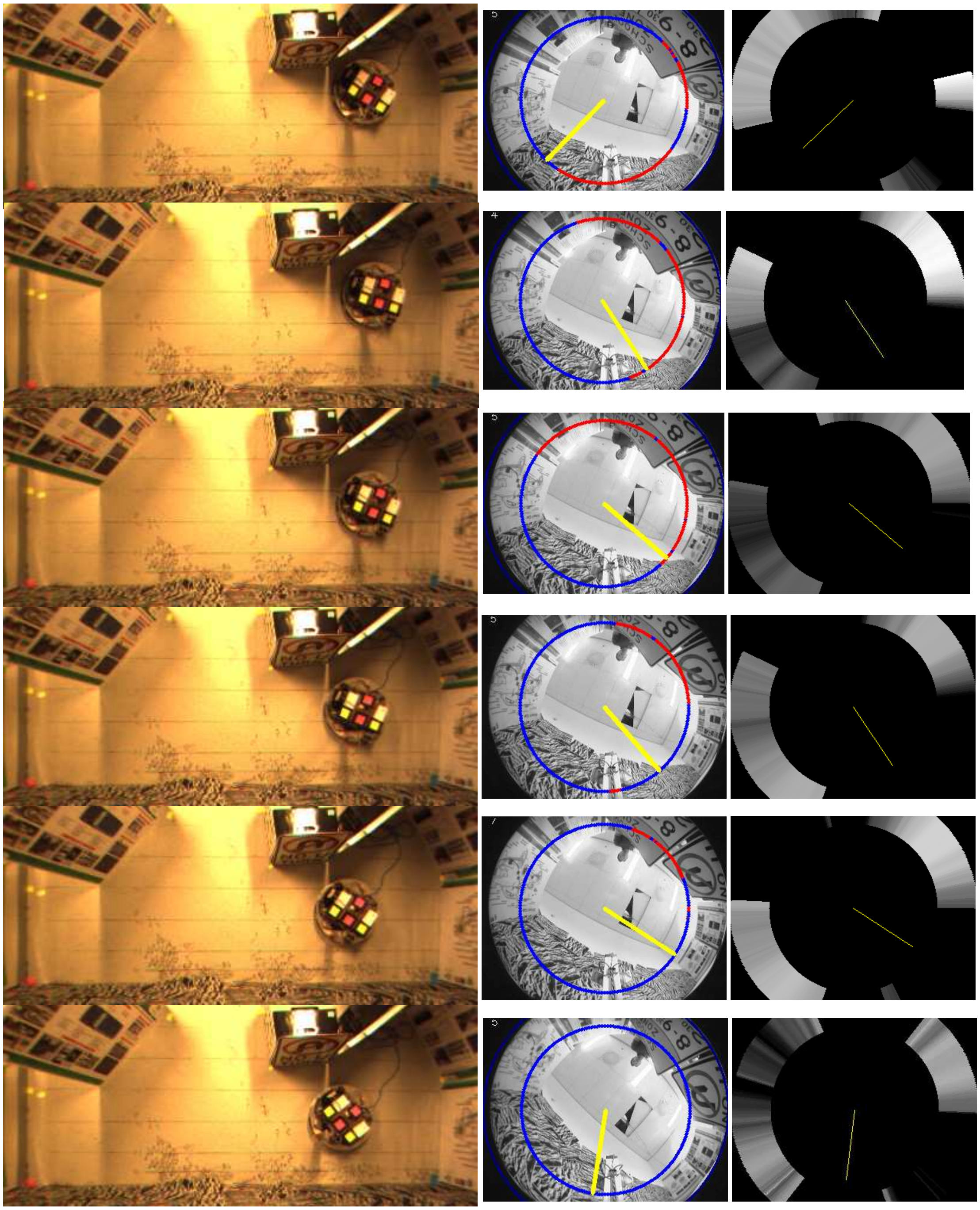

Figure 7.12: Sample overhead frames (left), onboard frames (middle) and intensity depth maps (right) from controlled environment experiment. Depth maps are computed in 120^o regions of interest on either side of the estimated translation direction (given by the yellow line). Increasing intensity represents the closer proximity of surfaces.

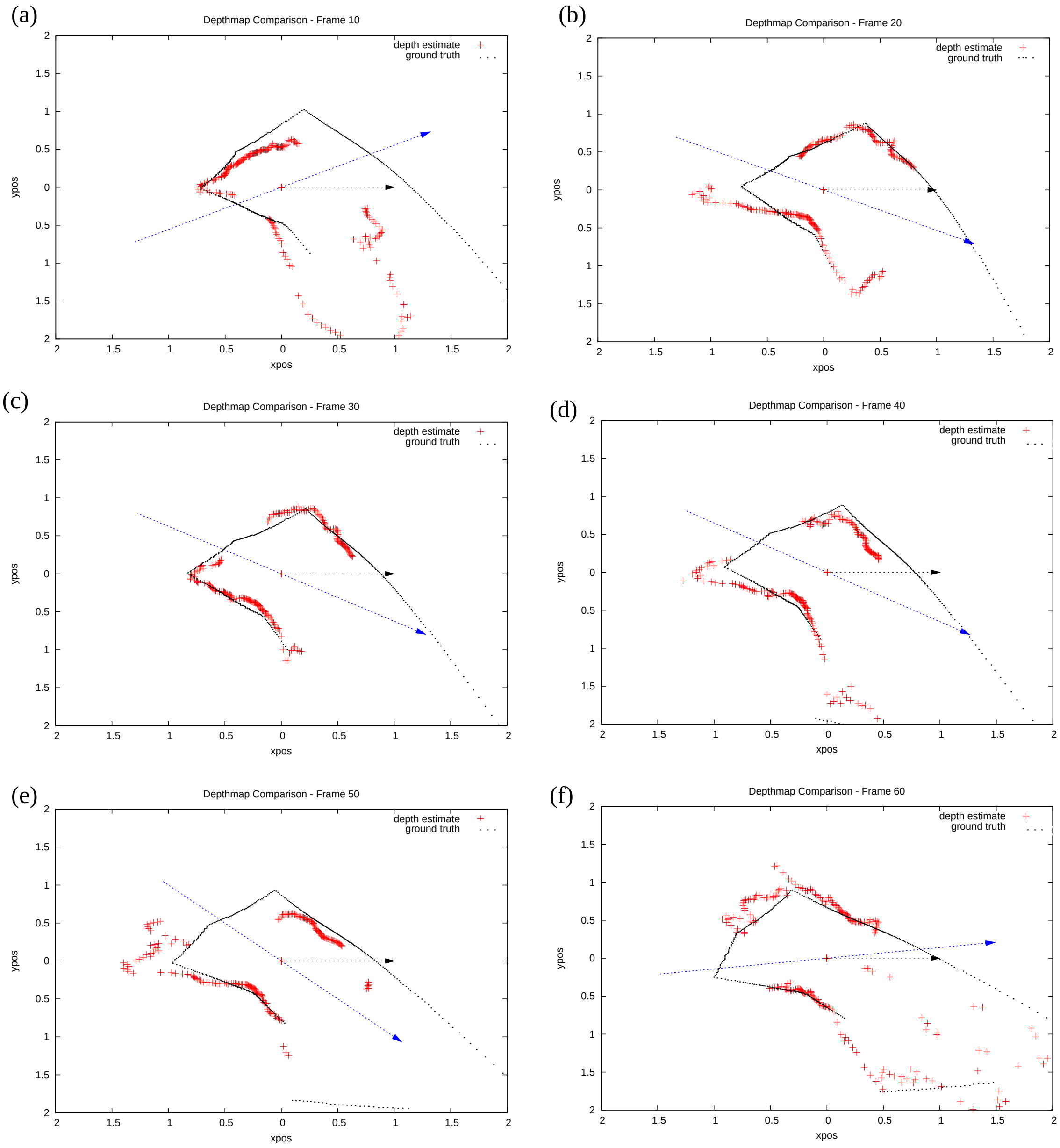

Figure 7.13: Sample depth maps (taken every 10 overhead frame intervals) for controlled environment experiment. Both estimated and ground truth depth values are scaled by the average depth of their respective depth maps. The short black arrow indicates the robot's orientation, which is always aligned with the X axis in the above figures. Note that an obstruction on the robot's hull obscures depth map estimation in this direction. The long blue arrow indicates the robot's estimated translation direction. Depth maps were computed across 120^o fields on either side of the translation direction.

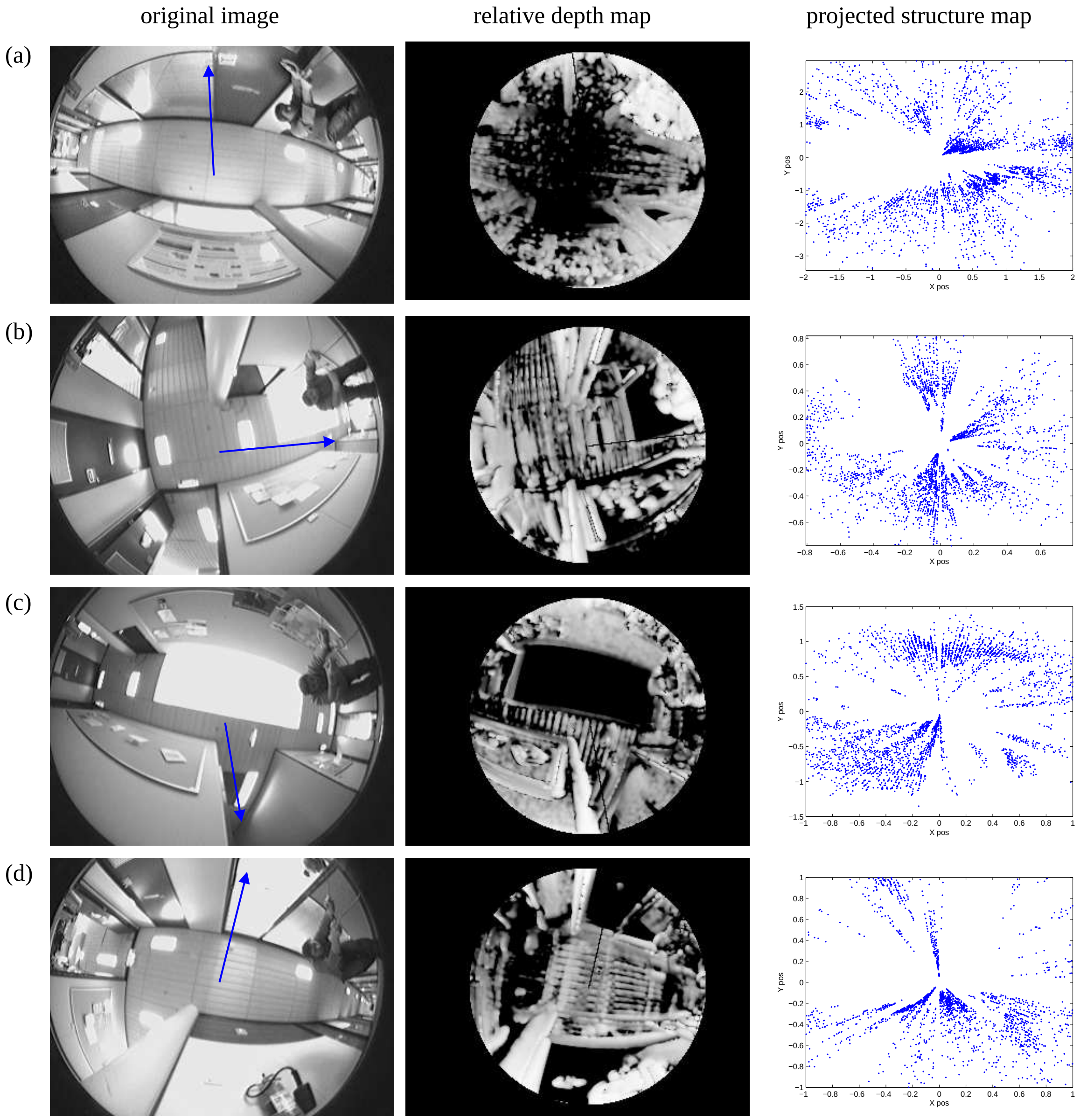

Figure 7.14: Sample depth maps obtained on-board the mobile platform (camera facing up). The left column shows the original image, and estimated direction of translation obtained from derotation. The middle column shows grayscale relative depth maps computed from the translational flow. The right column shows structure maps, obtained by projecting relative depth estimates into 3D space, and then orthographically onto the ground plane. Increasing brightness in intensity depth maps indicate closer proximity of surfaces.

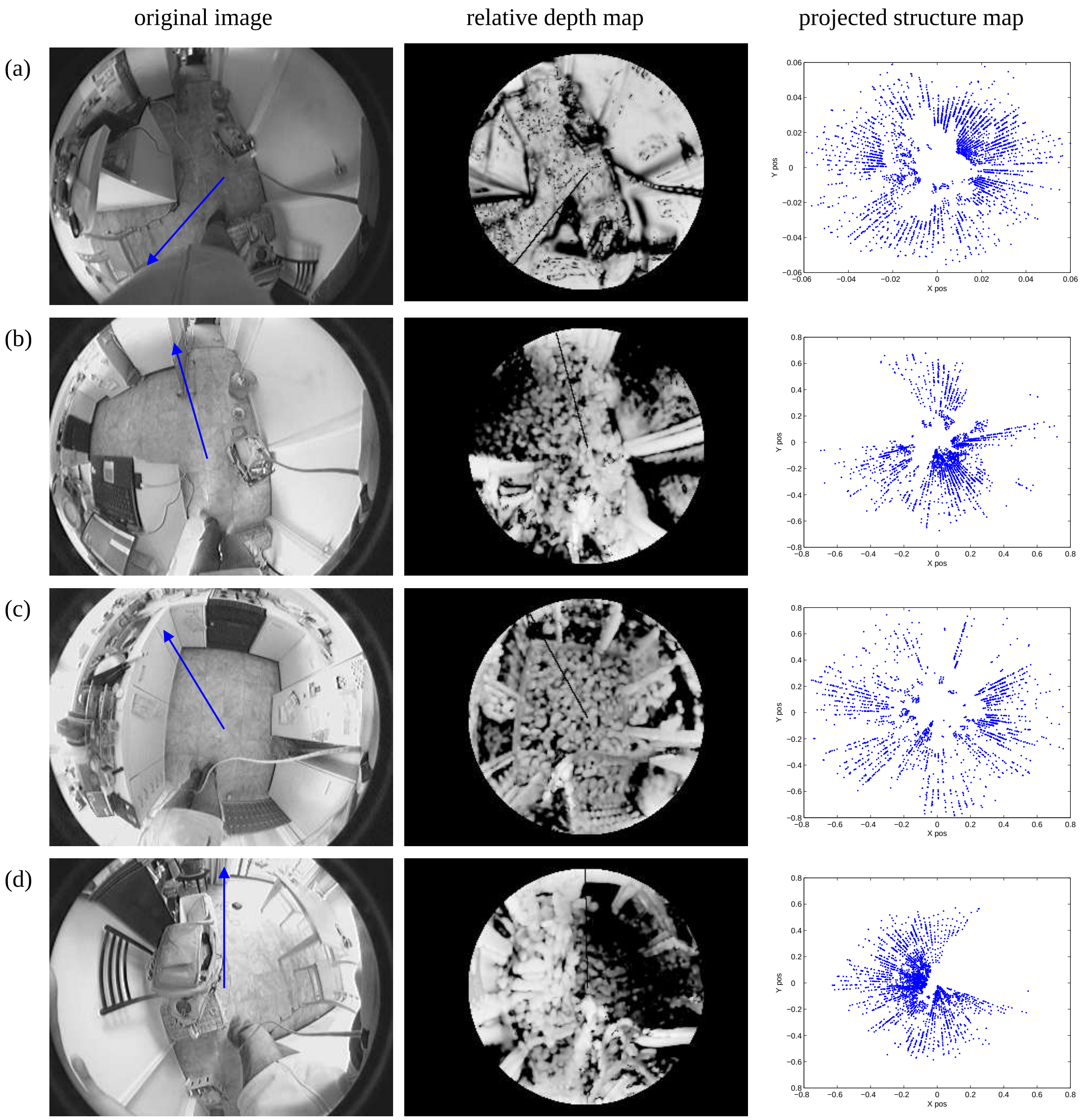

Figure 7.15: Sample frames and depth maps from the cluttered kitchen fly-through sequence (hand held camera facing towards ground plane). The left column shows the original image, and estimated direction of translation obtained from derotation. The middle column shows intensity depth maps computed from the translational flow. Increasing brightness indicates the closer proximity of surfaces. The right column shows structure maps, obtained by projecting relative depth estimates into 3D space, and then orthographically onto the ground plane.

mated from the de-rotated flow field. The third column provides a top-down view of the relative depths of scene points, projected onto the ground plane (we refer to these as structure maps). The centre of the structure map gives the location of the camera. For this, thresholding was applied to extract only the closest surfaces in the scene (and thus omit depth estimates from the ceiling). On a 2.1 GHz machine, depth maps were generated at rate of 1.2 updates per second over circular image regions of radius 110 pixels (*i.e.,* an area of approximately $38,000$ pixels).

The relative depth maps obtained over the corridor navigation sequence provide a good qualitative representation of the environment. An abundance of clear structural cues resulting from the motion of surface boundaries such as corners, doorways and windows can be seen. In addition, there appears to be good visual evidence of objects in close proximity being detected. This is particularly evident in Figures 7.14(c) and (d) where the wall edge in (c), and column (and fire hydrant) in (d) show up as the brightest areas in the intensity depth maps.

Structure maps in the third column of Figure 7.14 further support the accuracy of the relative depth measures for inferring basic scene structure. Most evident is the extraction of free space from obstructed space in the local area about the robot. This is evident in all samples. It is important to note, however, that space marked as unobstructed may also be the result of a lack of measurable flow in the area. Thus, some surface areas have only a few depth measures associated with them.

Notably, the sequence involves significant variation in lighting conditions as the robot travels beneath fluorescent lights, and past sun lit rooms. While optical flow estimates in these regions are generally unreliable, the wide field of view ensures enough features exist to extract overall scene structure, despite the noise inevitably introduced by these effects.

7.3.3.3 Cluttered environment fly-through results

Figure 7.15 shows results obtained for the cluttered kitchen fly through experiment. It can be seen that depth maps exhibit less structural definition than the corridor sequence, reflecting the relatively unstructured nature of the environment, and the greater abundance of objects in close proximity to the camera.

The camera's orientation towards the ground plane appears to significantly improve the extraction of free space from obstructed space. While evident in depth map results, this is made clearer in structure maps, particularly Figure 7.15(a), where the structure map provides a highly detailed map of free space over a considerable portion of the viewing area. In addition, the structure map shows an abundance of structural cues. Other samples from the sequence also exhibit clear and accurate extractions of free space.

These results are particularly encouraging when considering the camera was hand held and walked through the scene. While motion was predominantly in the horizontal plane, the camera was subject to both rotational motions off the plane, and changes in height throughout the sequence. Despite this, egomotion estimation appears to provide workable de-rotation for depth mapping in real world conditions.

7.4 Discussion

Qualitatively, results suggest the estimated depth maps provide a reasonable approximation of basic 3D depths in the scene. Across all image sequences, clear distinctions between free and obstructed space are obtained, and appear to be consistent with overall scene structure. Notably, ground truth comparisons suggest accuracy is reduced as surface depth increases. This is a likely result of the reduced flow magnitude generated from the slower apparent motion of distant surfaces. Increasing translational motion or image resolution should improve the depth estimates of distant surfaces. Alternatively, thresholding based on flow magnitude may also be applied, thereby ignoring distant surfaces posing no immediate threat of contact.

Egomotion estimation was generally observed to provide good robustness under varying camera motions and environmental conditions. This is made evident in ground truth comparisons from simulation and onboard testing. Notably, depth map results appear to reflect the underlying accuracy of egomotion estimation. For example, depth maps exhibiting highest correlation with ground truth in Figure 7.13 (*e.g.,* depth maps for frames 20, 30 and 40) generally correspond to accurate egomotion estimates in Figure 7.11. Conversely, less accurate depth maps in Figure 7.13 (*e.g.,* frames 50 and

60) correspond to periods of less accuracy in Figure 7.11. This egomotion estimation error appears most prevalent when rotational velocity is high with respect to translation velocity. Errors in translation direction are also most evident at these locations.

Qualitative results from the corridor and cluttered fly-through sequences suggest egomotion estimation is performing well over real-world images. Both sequences depict significant rotations, yet little ill-effects appear in the depth maps obtained. While no ground truth comparison is available for these sequences, it is evident from both image sequences, and the depth maps generated, that the algorithm provides sufficient accuracy to facilitate the real-time recovery of both the direction of ego-motion, and basic 3D structure.

Computing global 3D depth maps at full image resolution cannot be achieved in real-time at current CPU speeds without specialised hardware-specific programming (*e.g.,* GPU). For most navigation applications, however, this level of accuracy and coverage is not required. In the case of robot navigation, considerations of both the robot's physical height, and constraints on its motion may be exploited to limit the depth map generation field of interest. For typical structured environments, depth map resolution may also be reduced without significant impact on navigation support.

The Nelson and Aloimonos egomotion algorithm provides opportunities for significant speed-ups in execution. In particular, parameter choices associated with the search-based de-rotation of flow about each rotation axis. These choices include the search space size, as determined by the bounds on rotational velocities, and its quantisation. Reducing the size and/or resolution of this space can reduce execution time significantly. Naturally, this represents a trade-off of speed and accuracy. The loss of accuracy in depth maps obtained during high rotation in the onboard controlled environment experiment appears to reflect this trade-off. Notably, accuracy improves when translational motion is dominant. A possible improvement is to search over a non-uniform sampling of the rotation space, providing higher resolution for large rotations, thereby improving rotation estimation when rotation is most influential. Alternatively, where continuous depth map generation is not crucial or where motion is predominantly rotational, it may be appropriate to avoid depth map generation. However, this would not be a viable option if used as a direct input to motion control.

Overall, results suggest the depth map generation scheme is best suited to conditions where translational velocity is high. These conditions suit both the Nelson and Aloimonos egomotion estimation algorithm, which requires adequate translational velocity to find an even partition, and the accurate estimation of scene depth from local flow estimates.

7.5 Implications for general flow-based navigation

7.5.1 Structure-from-motion versus visuo-motor control

While results are encouraging, issues associated with the application of flow-based structure-from-motion algorithms for real-time contact estimation remain evident. To support tasks such as landing and docking, the scheme requires an adequate distribution of features in the scene to accurately estimate egomotion. However, it also requires sufficient density of features within local image patches to support motion control in close proximity with surfaces. Visuo-motor based approaches typically require only the latter, and may actively select regions within the image that provide the best possible measure of proximity with surfaces. For example, the divergence maximum or FOE provide task-meaningful locations from which to measure surface proximity for landing and docking. While structure-from-motion techniques may exploit features such as the FOE (for egomotion estimation), they lack the context of task in which to apply such cues more usefully.

Efficiency concerns also remain prevalent for full egocentric depth map recovery. While the scheme presented provides efficient recovery of dense global structure, it is unlikely to provide adequate support for operation in the control loop of tasks such as landing and docking. For navigation subsystems in direct control of motion, it makes sense to revert to visuo-motor control.

7.5.2 Towards systems of visual control

Visuo-motor schemes such as those presented in this thesis require specific visual conditions to exist prior to their invocation. Structure-from-motion recovery may therefore be used to monitor conditions, and invoke appropriate visuo-motor behaviours when environmental priors are satisfied. For example, use of the divergence maximum to guide a landing manoeuvre assumes the max-div point lies within the projected area of a target planar surface. Structure-form-motion recovery may therefore be used to identify planar surfaces in the scene, select the target surface and invoke the max-div scheme. Estimates of surface orientation parameters may also be used to assist visuo-motor approach angle regulation, which as discussed in Section 6.3.2, is under constrained using the max-div point alone. Defining such a role for structure-from-motion removes it from the control loop, thereby alleviating the need for highly accurate, rapidly obtainable solutions. Under this framework, structure-from-motion techniques such as that presented in this chapter can be effectively applied. We note, however, that classically derived structure-from-motion techniques such as [Nister 2005] and [Pollefeys et al. 2008] offer potential real-time structure-from-motion alternatives, providing increasing structural detail. Thus, classical structure-from-motion in the control loop will soon be possible, and for certain applications, may provide a viable alternative to the biologically-based approaches described in this thesis.

7.6 Summary

We have presented a strategy for generating 3D relative depth maps from optical flow in real-time. In so doing, we have demonstrated for the first time, the use of the Nelson and Aloimonos egomotion algorithm over real images, depicting real-world environments. Results from simulated full general motion of a sphere, and from a series of real-world experiments suggest this strategy may be a useful base for many navigational subsystems. The results, however, suggest its use is most appropriate for high-level navigation tasks where accuracy and efficiency is less critical.

In addition, these results further support theoretical arguments in favour of a spherical projection when attempting to infer scene structure and self-motion from optical flow. This, in conjunction with advantages identified for time-to-contact estimation in Chapter 6, provide compelling support for the use of a spherical projection model for flow-based navigation and perception.

We have now examined the two dominant approaches to visual contact estimation for robot navigation: via direct perceptual visuo-motor control schemes, and through the recovery of full structure-from-motion. In the remaining chapters we consider visual contact estimation in the context of self-moving objects. We present preliminary work in the development and application of new flow-based visual cues for threat perception and avoidance.

Chapter 8

Time and location of impact prediction based on primate vision

8.1 Introduction

In the previous chapters we have considered visual contact estimation in the context of robot navigation and control. These techniques, however, are designed for use in a predominantly static environment. Another important capability of any system (robotic or biological) working in a dynamic environment is the ability to perceive potential contact with independently moving objects. This is particularly important when the object poses an imminent threat of collision with the observer. Detection of the threat must occur with sufficient time to allow evasive actions to be taken. This may involve movement away from the threat to avoid collision, or the invocation of motor responses to intercept its course (*e.g.,* catching or deflecting prior to impact). In either case, the ability to predict where a potential threat exists, and where it is heading, is essential.

In this chapter we explore the use of optical flow to realise both capabilities. We present preliminary work examining the use of optical flow, under spherical projection, to predict the time and location of impact of an incoming object about a stationary camera. The proposed contact estimation scheme is modelled on the observed behaviour of bi-modal neurons in the F4 region of the pre-motor cortex in primates.

The chapter is structured as follows. Section 8.2 presents the motivations and a brief discussion of relevant background literature. Section 8.3 presents our approach.

Section 8.4 describes preliminary experiments conducted and results obtained. Section 8.5 provides a discussion of these results and future directions for this research. Finally, Section 8.6 provides a chapter summary.

8.2 Background and motivation

Neuroscience researchers have recently identified specific neurons in the pre-motor cortex of primates specifically responsible for the sensory guidance of reaching and intercepting incoming objects [Fogassi et al. 1996]. Evidence suggests neurons in the F4 region of the pre-motor cortex are critically involved in the control of reaching movements. Area F4 contains a representation of head, torso and arm movements. A subset of these neurons, referred to as *bimodal neurons*, have been shown to be spatially mapped to regions about the upper body [Fogassi et al. 1996]. These neurons discharge when contact is made with the region of the body to which they are mapped. Notably, however, the same neural response is generated when visual stimuli in the 3D space adjacent to the mapped body region suggests impact is likely to occur at that location. Moreover, the depth of the receptive field of these neurons has been shown to be sensitive to the speed of approaching stimuli, allowing faster moving objects to be detected earlier [Fogassi et al. 1996]. This suggests the apparent motion of the object, as represented by the optical flow, may provide the primary visual cue upon which predictions of the impact location are based. It has been proposed that this neural encoding is achieved by associating specific patterns of flow with corresponding localised tactile stimuli. These associations are most likely learnt in the early years of development (See Zako *et al.* [2009] for a recent review of neuropsychological studies on trajectory estimation and interception in humans).

Computationally, it is of interest to consider the use of optical flow as a primary cue for predicting the course of incoming objects. A fundamental question, however, is whether current optical flow estimation techniques provide adequate support for such applications. To obtain fast impact predictions, and to maintain relevance to the underlying biological model, we compute time and location of impact from instantaneous local patterns of optical flow (*i.e.,* we do not perform temporal integration). We therefore regard this as an impact detector, upon which more complex systems may be built.

8.2.1 Previous work

The detection of independently moving objects in a scene has been an active field of research in computer vision for some time. Given two or more time-separated views of the scene, the problem becomes one of motion segmentation, whereby regions of homogeneous motion are grouped together. Geometric methods such as those described by Li [2007] and Vidal and Hartley [2004], attempt to optimally recover the motion of all objects in the scene from matched points in two or more views. While often highly accurate, these techniques are currently not feasible for real-time use. Other approaches examine the apparent motion of objects in the scene via explicit tracking of feature points [Yamaguchi et al. 2006; Cohen and Medioni 1998], or via the optical flow field generated by the motion of objects [Mitiche and Sekkati 2006; Weber and Malik 1997]. A particularly challenging problem is the segmentation of optical flow due to independently moving objects from flow due to self-motion. Despite much attention, the problem remains difficult, and an active area of research in computer vision.

In Chapter 4 we extensively reviewed the literature on estimating time-to-contact with looming objects from local flow field differential invariants. Such approaches provide a direct means of estimating the rate of approach of objects without explicitly computing the object's motion parameters. Such work, however, does not attempt to predict the location of impact of incoming objects.

Numerous active vision and tracking systems have been proposed for estimating the trajectory of moving objects in the scene. Kundur and Raviv [1999] obtain proximity estimates for incoming objects from the measured blur of fixated texture-patches. No prediction of impact location is provided, however, time varying changes in fixation direction should provide some indication of this. Hong and Slotine [1997] describe active vision systems for tracking and catching tossed balls. While active vision systems such as these support object interception and threat avoidance, they require tracking over time. They do not make predictions from instantaneous visual cues.

The work presented in this chapter is similar to research by Ogino *et al.* [2006], who make use of a neural network to predict the arrival time and location of an incoming ball from learnt patterns of visual motion. Using a humanoid soccer playing robot, the neural network is trained to identify different trajectories of the ball, from which a motor response to trap the ball with the robot's foot is invoked. The ball is first identified in the image, from which the causal relationship between the ball's position and the optical flow is learnt. The emphasis of this work is on learning globally defined patterns of optical flow over a wide field of view to predict the ball's incoming trajectory.

The prediction scheme we propose here considers the underlying use of optical flow in realising such behaviours. In contrast to Ogino *et al.* [2006], we do not attempt to explicitly predict a trajectory or learn patterns of motion associated with trajectory data. We base predictions of time and location of contact on the examination of local instantaneous flow field patterns resulting from the motion of arbitrary incoming objects. We do not consider the use of machine learning techniques, temporal filtering or tracking. Instead, we focus specifically on how spatial differential properties of the instantaneous flow field may be exploited to predict time and location of impact.

8.3 Proposed method

As stated, the current work aims at verifying the quality of measurements obtained from optical flow. In so doing, we make the following assumptions: a stationary camera, predominantly translational motion of a single incoming object, and a roughly planar surface facing towards the camera. We derive the scheme under a spherical projection model. However, a narrow field of view about the optical axis should also provide adequate results. Given object fixation (such as in primate vision), such an assumption is reasonable.

We seek to predict the relative impact location of an incoming object on a planar body centred on the image origin (*i.e.,* the projective centre of the image). Thus, we may regard the planar body as the infinite extension of the image plane in all directions. We refer to this plane as **B**.

The approach adopted applies principles introduced in Chapter 6. Given optical flow under spherical projection, we consider both the optical flow field, and the flow field divergence for a local tangent plane. Assuming translational motion only, we omit the rotational component of Equation 3.7 to obtain:

$$f(\hat{p}) = \frac{-v}{R(\hat{p})}\left((\hat{p} \cdot \hat{t})\hat{p} - \hat{t}\right), \tag{8.1}$$

where $\hat{p} \in \mathbb{R}^3$ is the viewing direction, $\hat{t} \in \mathbb{R}^3$ is the direction of object translation with respect to the camera and v, its velocity, and $R(\hat{p})$ is the depth of the surface point in the viewing direction.

Let $\hat{q} \in \mathbb{R}^2$ be the direction of the object's translational motion in the tangent plane to $\hat{p}$ on the view sphere. We wish to obtain the component of motion in this direction, such that:

$$f(\hat{p}) \cdot \hat{q} = \frac{-v}{R(\hat{p})}\left((\hat{p} \cdot \hat{t})(\hat{p} \cdot \hat{q}) - (\hat{t} \cdot \hat{q})\right). \tag{8.2}$$

Noting that $\hat{q}$ is perpendicular to $\hat{p}$, we simplify the above to:

$$f(\hat{p}) \cdot \hat{q} = \frac{v(\hat{t} \cdot \hat{q})}{R(\hat{p})}. \tag{8.3}$$

We consider now the flow field divergence. For convenience, we recall the divergence equation on the view sphere:

$$\texttt{div}(\hat{p}) = \frac{v(\hat{p} \cdot \hat{t})}{R(\hat{p})}\left[1 + \frac{\Delta R(\hat{p})}{R(\hat{p})}\Big(\frac{\hat{t}}{(\hat{p} \cdot \hat{t})} - \hat{p}\Big)\right]. \tag{8.4}$$

As discussed in Section 3.3.2, the presence of the unknown depth gradient term, $\Delta R(\hat{p})$, introduces a deformation component into the divergence estimation, denying a precise estimate of time-to-contact. It can be seen, however, that the depth variation term is scaled by the distance of the object. Thus, if the object is sufficiently far away, the contribution of this component is likely to be small. We therefore make the assumption that depth variation will be small with respect to the distance of this variation from the sensor, and set the depth gradient term to zero. As a consequence, significantly inclined surfaces may result in less accurate predictions of impact time and location.

Setting $\Delta R(\hat{p}) = 0$, we reduce the divergence equation to:

$$\texttt{div}(\hat{p}) = \frac{v(\hat{p} \cdot \hat{t})}{R(\hat{p})}, \tag{8.5}$$

and thus the time-to-contact along the viewing direction $\hat{p}$ is given by:

$$\tau_{\mathrm{p}} = \frac{R(\hat{p})}{v(\hat{p} \cdot \hat{t})}. \tag{8.6}$$

8.3.1 Computing angle of approach

We seek to estimate the angle of approach of the object with respect to the viewing direction, $\hat{p}$. This may then be used to obtain a constraint on the location of impact within the planar body **B**. Dividing Equations 8.3 and 8.5 we obtain the ratio of motion parallel to the tangent plane of $\hat{p}$, and the component of motion along $\hat{p}$, such that:

$$\frac{f(\hat{p}) \cdot \hat{q}}{\texttt{div}(\hat{p})} = \frac{(\hat{t} \cdot \hat{q})}{(\hat{t} \cdot \hat{p})}. \tag{8.7}$$

Thus, the angle of approach with respect to $\hat{p}$ is defined as:

$$\theta_p = \tan^{-1}\Big(\frac{(\hat{t} \cdot \hat{q})}{(\hat{t} \cdot \hat{p})}\Big). \tag{8.8}$$

8.3.2 Estimating location of impact

We now compute the location of impact in **B**. Specifically, we compute the direction $\hat{u} \in$ **B**, and the scaled radial distance d_u of the impact point, I, with respect to the centre of **B** (*i.e.,* the origin, O). To assist these derivations, we first define the plane containing O, P, and the yet to be determined I (see Figure 8.1). Note that this plane contains $\hat{t}$, and thus defines the object trajectory plane. We refer to this triangular region as **OPI**.

8.3.2.1 Direction of impact

We first consider $\hat{u}$. The line defined by $\hat{u}$ in **B** is given by the intersection of **OPI** with **B**. Thus, given knowledge of $\hat{p}$ with respect to B (from camera calibration), and

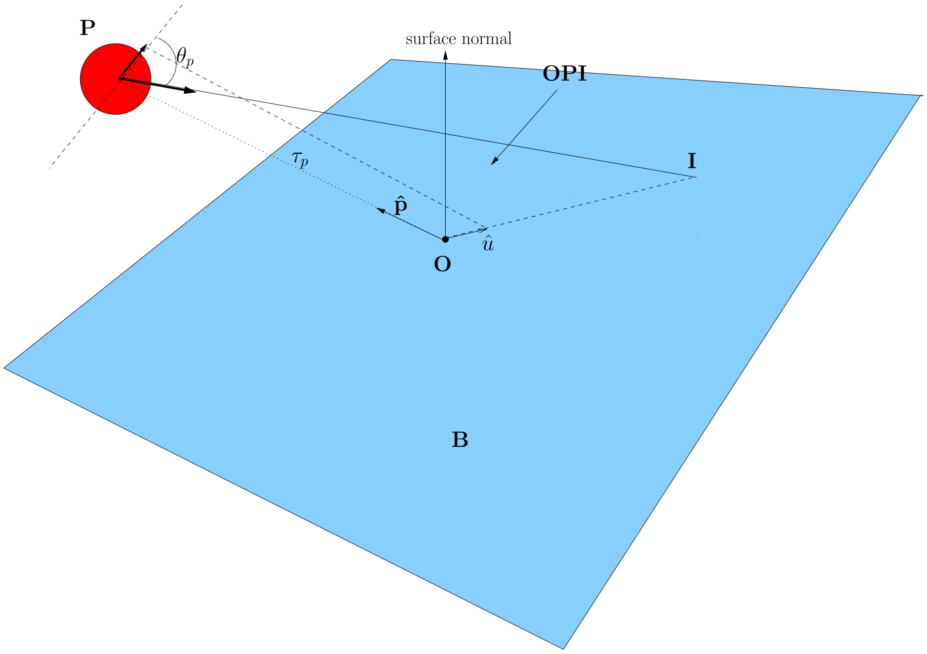

Figure 8.1: Geometric representation of impact point (I) prediction of an incoming object P, on the planar body **B**. O is the camera-centred origin.

$\hat{q}$ from the optical flow, $\hat{u}$ may be determined by translating $\hat{q}$ along $\hat{p}$ to the origin, and projecting it into **B**. This intersection defines a line of possible impact locations (see Figure 8.1). We refer to this line as the *line of impact*.

8.3.2.2 Camera-centred location of impact

To recover d_u, it is necessary to first determine the current projected location of P on the infinitely extended line defined by $\hat{u}$. Note that $\hat{u}$ exists in both **B** and **OPI**, thus allowing d_u to be recovered without regard for the relative orientation of **B** and **OPI**. We therefore consider only the plane of **OPI** in the remainder of this derivation. Figure 8.2 shows this graphically.

Let $\hat{n} \in \mathbb{R}^2$ define a vector orthogonal to $\hat{u}$ in **OPI**, and ϕ be the angle between the directions $\hat{p}$ and $\hat{n}$. Let N denote the current projected location of P on the line of impact (*i.e.*, the projection of P along $\hat{n}$), and d_n be the scaled radial distance of N

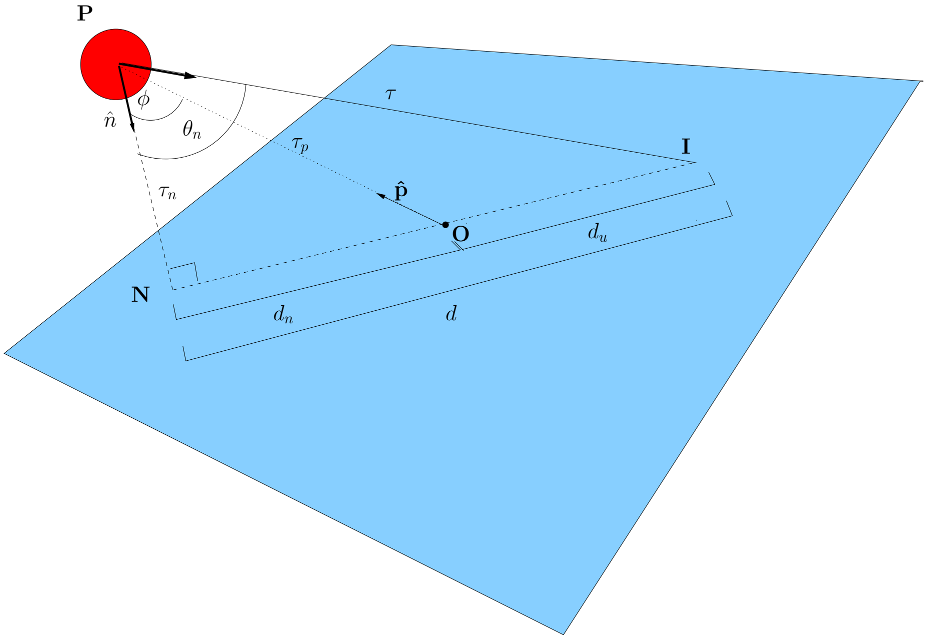

Figure 8.2: Figure shows geometric relationships exploited to predict the scaled distance of the point of contact along the line of impact with respect to the origin.

from the the origin. We solve for d_n by projecting τ_p onto the line impact such that:

$$d_n = \tau_p \sin(\phi) \tag{8.9}$$

Let d be the total scaled distance between N and I, and θ_n be the angle of approach of the object with respect to $\hat{n}$, such that:

$$\theta_{\mathrm{n}} = \theta_{\mathrm{p}} + \phi, \tag{8.10}$$

To solve for d we first project τ_{p}, along $\hat{n}$ such that:

$$\tau_{\mathrm{n}} = \tau_p \cos\phi, \tag{8.11}$$

from which we obtain:

$$\begin{aligned} d &= \tau_{\mathrm{n}} \tan\theta_{\mathrm{n}}, \\ &= \tau_p \cos\phi \tan\theta_{\mathrm{n}}. \end{aligned} \tag{8.12}$$

Given d_n and d, we may easily determine the radial distance, d_u:

$$d_u = d - d_n. \tag{8.13}$$

Expressing this in terms of directly measurable (or known) quantities:

$$d_u = \tau_{\mathrm{p}}\Big(\cos\phi\tan(\theta_{\mathrm{p}} + \phi) - \sin(\phi)\Big). \tag{8.14}$$

As can be seen, the distance, d_u, is defined as a proportion of τ_{p}. Thus, the estimated radial distance of impact is defined in temporal units, scaled by the measured time-to-contact along the viewing direction.

8.3.3 Estimating time-to-impact with the plane

To infer the incoming object's time-to-impact, τ, with I we compute the temporal distance along the angle of approach. This can be obtained directly from τ_{n} and d such that:

$$\tau = \sqrt{\tau_n^2 + d^2} \tag{8.15}$$

Substituting for τ_n and d, we obtain:

$$\begin{aligned}\tau &= \tau_p\sqrt{\cos^2\phi + (\cos^2\phi\tan^2(\theta_{\mathrm{p}} + \phi))},\\ &= \tau_p\cos\phi\sec(\theta_p + \phi). \end{aligned} \tag{8.16}$$

8.4 Performance assessment

8.4.1 Implementation

The algorithm is applied in the following steps:

1. Identify regions of movement in the image, and compute flow divergence across region.

2. Compute time-to-contact with respect to the camera (*i.e.*, τ_p) at the divergence maximum within the segmented region.

3. Estimate the angle of approach of the object (discussed further below).

4. Compute location and time-of-impact with the camera plane via Equations 8.11, 8.14, and 8.16.

The first step of the algorithm identifies regions in the image where object motion exists. This is achieved using simple image differencing of image intensity values, and applying an appropriate threshold to account for noise. The optical flow within each region is then used to compute the divergence, from which the region of maximum divergence is selected. In addition, the segmentation strategy grows regions of motion where positive flow divergence is above a set threshold, and spatially consistent. Thresholding is performed on the average divergence computed in 5×5 pixel regions. Neighbouring regions above the threshold are then combined. Regions of negative divergence are ignored.

The most crucial step in algorithm is the estimation of the angle of approach. We achieve this by first estimating the predominant direction of motion of the segmented object in the image plane. Taking the point of maximum divergence, we convolve eight discrete templates of unit flow vectors, each depicting a discrete direction of motion in the image plane (*i.e.,* up, down, left, right, up-left, up-right, down-left, down-right). Based on the relative scores, a direction of most support is identified as the direction of tangential motion.

To estimate the angle of approach with respect to the viewing direction, we divide the magnitude of the optical flow in the identified dominant translational direction (as given by the score), by the average of divergence estimates taken within the same region. We then compute θ_{p} directly from this ratio (Equation 8.8), and from this, the angle of approach with respect to the objects current location on the line of impact, θ_n (Equation 8.10). Time-to-impact with the plane is computed using Equation 8.16.

8.4.2 Initial controlled-conditions testing

We first tested a basic implementation of the impact prediction scheme to assess the accuracy of approach angle estimates, and the impact location. For this, we assume the object of interest lies along the camera's viewing direction. We thus skip the first step of the algorithm, and compute the impact point for the surface projecting to the image centre.

The modified algorithm was run over two image sequences. Both sequences depict the controlled motion of a camera descending towards a planar surface, along a predetermined angle of approach, and at constant velocity. Note that these *landing sequences* are the same as used in Chapter 6. We tested the algorithm over sequences depicting a 22.5^o and 67.5^o approach, thereby providing test cases for a near frontal and significantly angled object trajectory. Figure 8.3 shows sample frames, and the central region used for estimating angle and location of impact.

Figure 8.4 shows results obtained for the estimated angle of approach of the surface in the central image patch. In both cases, a reasonable approximation to ground-truth is obtained. The 67.5^o approach deviates further from the ground truth, however, these estimates are more consistent than the 22.5^o approach. It should be noted that the angles of approach reported for both sequences are subject to some error due to the imprecise nature of the image construction process.

Figure 8.5 shows the estimated location of impact for each frame of both sequences. The origin of the coordinate system is the camera location. The location of impact is given in units of the estimated time-to-contact of the surface towards the camera. Thus, we expect the location to remain constant across each sequence. The dashed lines indicate the ground truth horizontal distance of the impact point from the camera. It is evident that the impact prediction is significantly more accurate for the 22.5^o approach than the 67.5^o approach. However, both sets of results achieve reasonable consistency in the estimated location.

To assess the accuracy of time-to-contact estimates with the impact location, we compare the ratio of time-to-contact estimates obtained along the direction of motion, and along the camera's central axis. This allows an assessment of accuracy against ground truth without the need for absolute values of time-to-contact, which are subject to scale factor differences. Accurate time-to-contact estimates should yield a constant ratio across the sequence. Figure 8.6 shows the time-to-contact ratio results. It is clear that high accuracy and consistency is achieved across the 22.5^o trial. Consistency and accuracy degrade for the 67.5^o trial, although in general, the estimated ratios remain

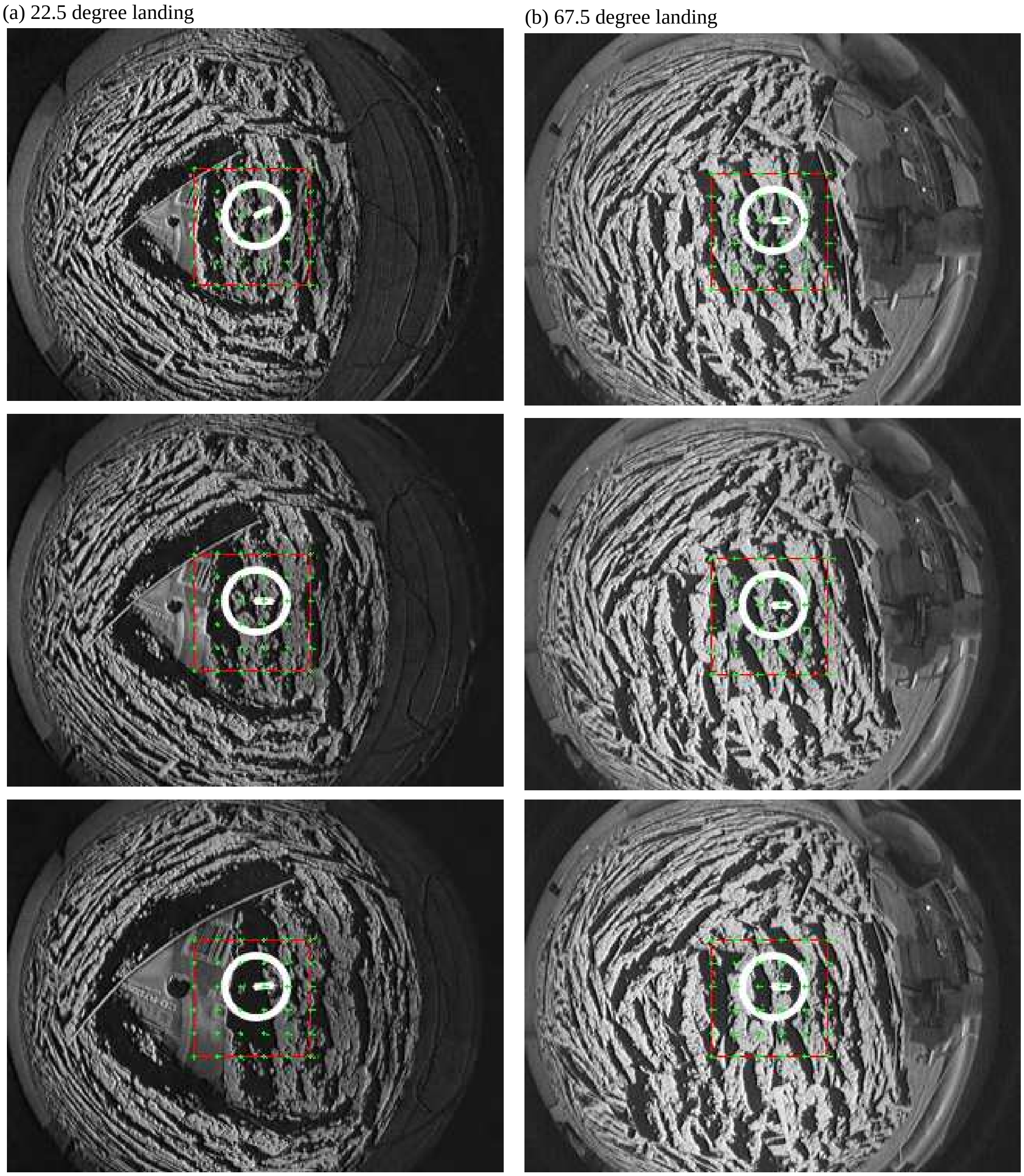

Figure 8.3: Sample frames from both the 22.5^o and 67.5^o landing sequences, showing the central region for which the angle of approach, and impact location are estimated. The camera's known angle of approach and constant velocity towards the surface provides ground truth data for comparison.

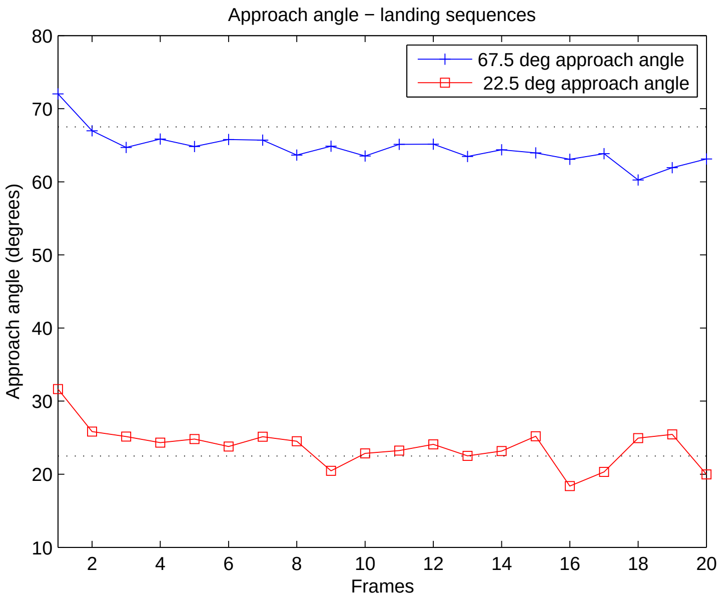

Figure 8.4: Estimated angles of approach for landing sequences depicting a 22.5^o and 67.5^o approach. Dotted line indicates ground truth for both approach angles.

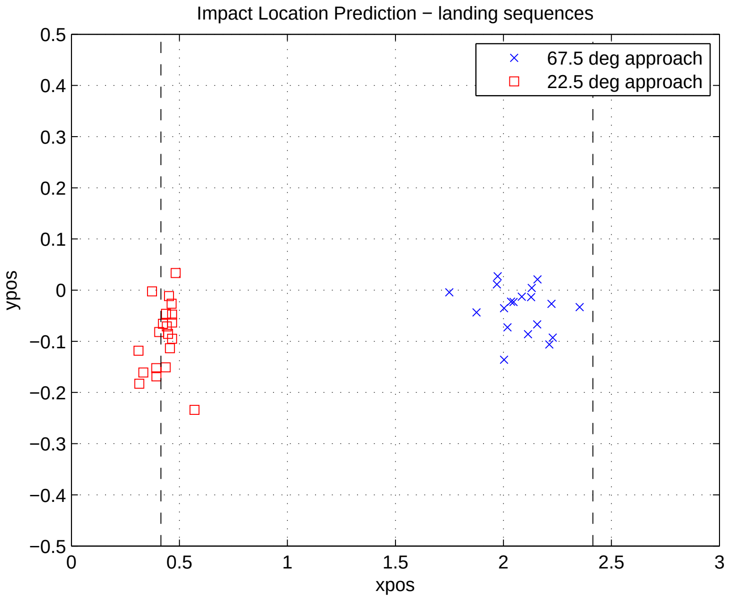

Figure 8.5: Estimated location of impact for each frame of the 22.5^o and 67.5^o landing sequences. Note that the camera centre is located at the origin of the coordinate system (middle left of graph). The predicted location of impact is given in units of the surfaces time-to-contact with the camera. The dashed line indicates the ground truth horizontal distance of the impact point from the camera centre.

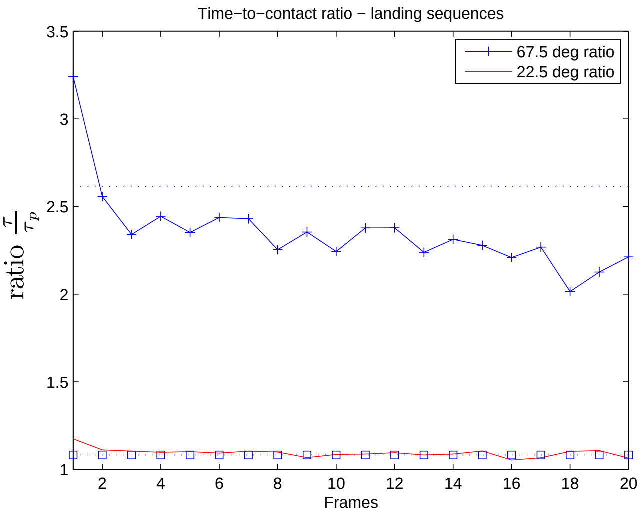

Figure 8.6: The estimated ratio of object time-to-contact with the infinite image plane along the direction of motion (τ), and along the viewing direction (τ_p). The ground truth ratio is also shown, as determined from the camera's known angle of approach. We compare ratios to avoid scale factor differences in absolute time-to-contact estimates.

stable.

8.4.3 Live impact prediction test

The full algorithm was implemented for real-time use with a stationary digital camera. Frames captured from the camera were fed directly into the impact prediction algorithm. The predicted impact point, and time-to-contact were both computed for each frame, and displayed in an adjacent image. The output of a live test of the system was written to a video file, and is included in the supplementary CD-ROM to this thesis (Video 8.1). Figures 8.7 and 8.8 show sample frames from the video output, showing both the detection of the incoming object (in the left image), and the predicted location of impact about the plane surrounding the camera (right image). Impact predictions for an object of interest are represented by rectangular markers, and are shown for all predictions prior to and including the current frame. Increasing brightness in impact point markers represents decreasing time-to-contact.

Sample output presented in Figures 8.7 and 8.8 show the development of impact location predictions as the object of interest approaches the impact plane. In both cases, early predictions appear to be outliers compared with later predictions, which appear to stabilise on a local region of probable impact. The accuracy of impact predictions appear stronger for the open hand than for the closed-fist, suggesting the approximate planar surface of the open hand yields better predictions. This is unsurprising given the algorithm assumes depth variation within the region of interest is zero. This is clearly not the case for the closed fist, where it can be seen that the max-div location (about which impact predictions are made) exhibits significant depth variation. This variation is likely to contribute to the divergence in the region, thereby biasing impact location prediction towards the viewer direction. This effect appears evident in later impact predictions for the closed-fist example. The fist's trajectory suggests impact markers should be further to the left of image centre than shown in the right column of Figure 8.8.

8.5 Discussion

These preliminary results provide quantitative and qualitative support for the feasibility of the proposed impact prediction algorithm. Results obtained across the experiments suggest that while accuracy may degrade for impact locations away from the camera, they remain stable and consistent. On this basis, the scheme appears to provide workable accuracy, suitable for a wide range of approaching object trajectories.

Impact predictions for more eccentric approach angles appear to be biased towards the direction of the object's origin. This is evident for the 67.5^o landing sequence, as well as the punching sequence in Figure 8.8. While more image sequences are needed to be conclusive about this, such a bias is consistent with results reported from human trials [Neppi-Mòdona et al. 2004], where subjects were asked to predict the location of impact of looming visual stimuli about their face. Similar assumptions to those stated here were applied in the experiments conducted. It is unclear from these preliminary experiments what the cause of this bias is. One possibility is that the estimated motion towards the camera is disproportionately represented by the divergence. The contributions of local deformation due to the apparent motion of non-fronto-parallel surfaces can increase the divergence estimate. Neppi-Modona *et*

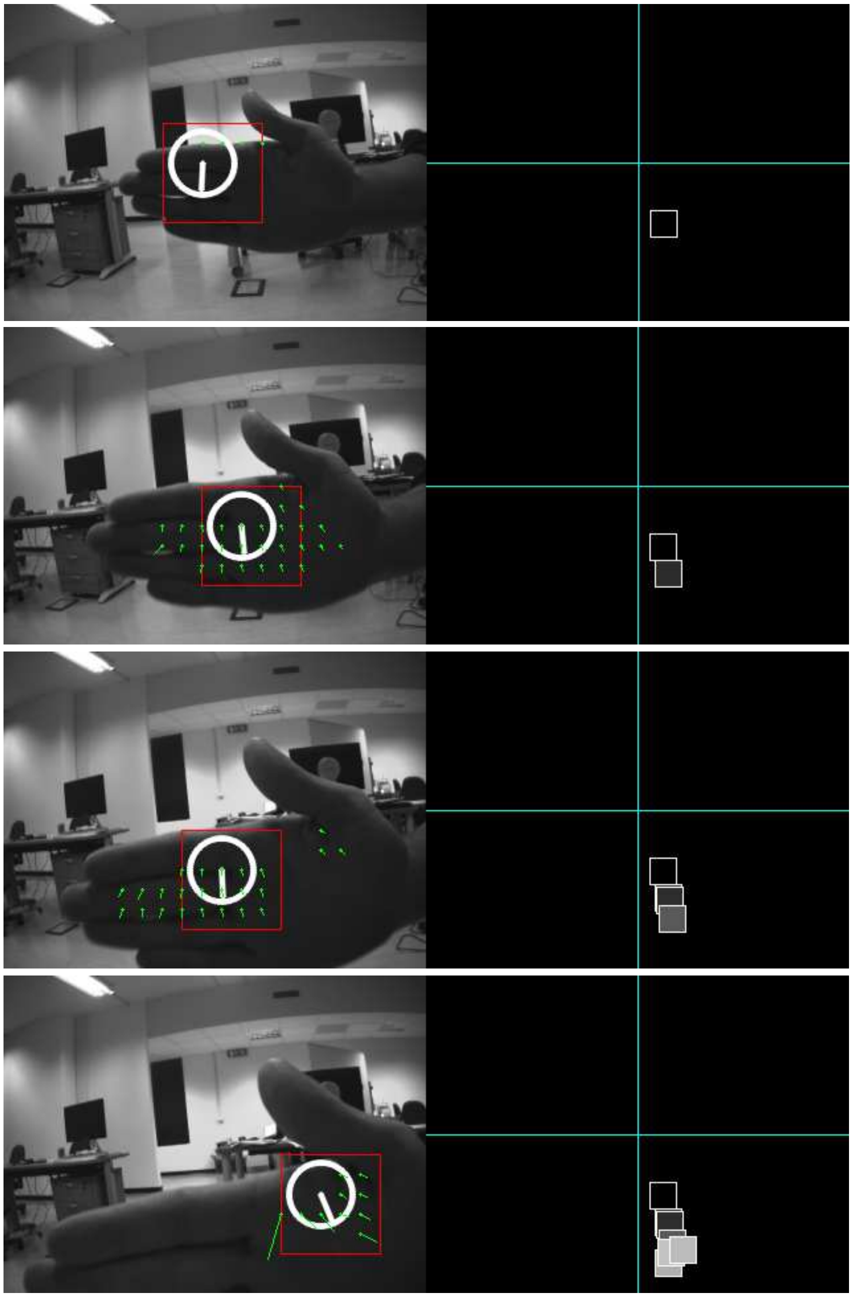

Figure 8.7: Sample output frames from live impact prediction test for an incoming open hand. Left image shows the optical flow in the segmented region of motion, and the estimated direction of translational motion in the plane. The red box represents the region of maximum divergence. Right image shows estimated location of impact about the camera. Increasing brightness in impact markers represents the estimated time-to-contact. Complete output from the impact prediction algorithm is given in Video 8.1 of the supplementary CD-ROM to this thesis.

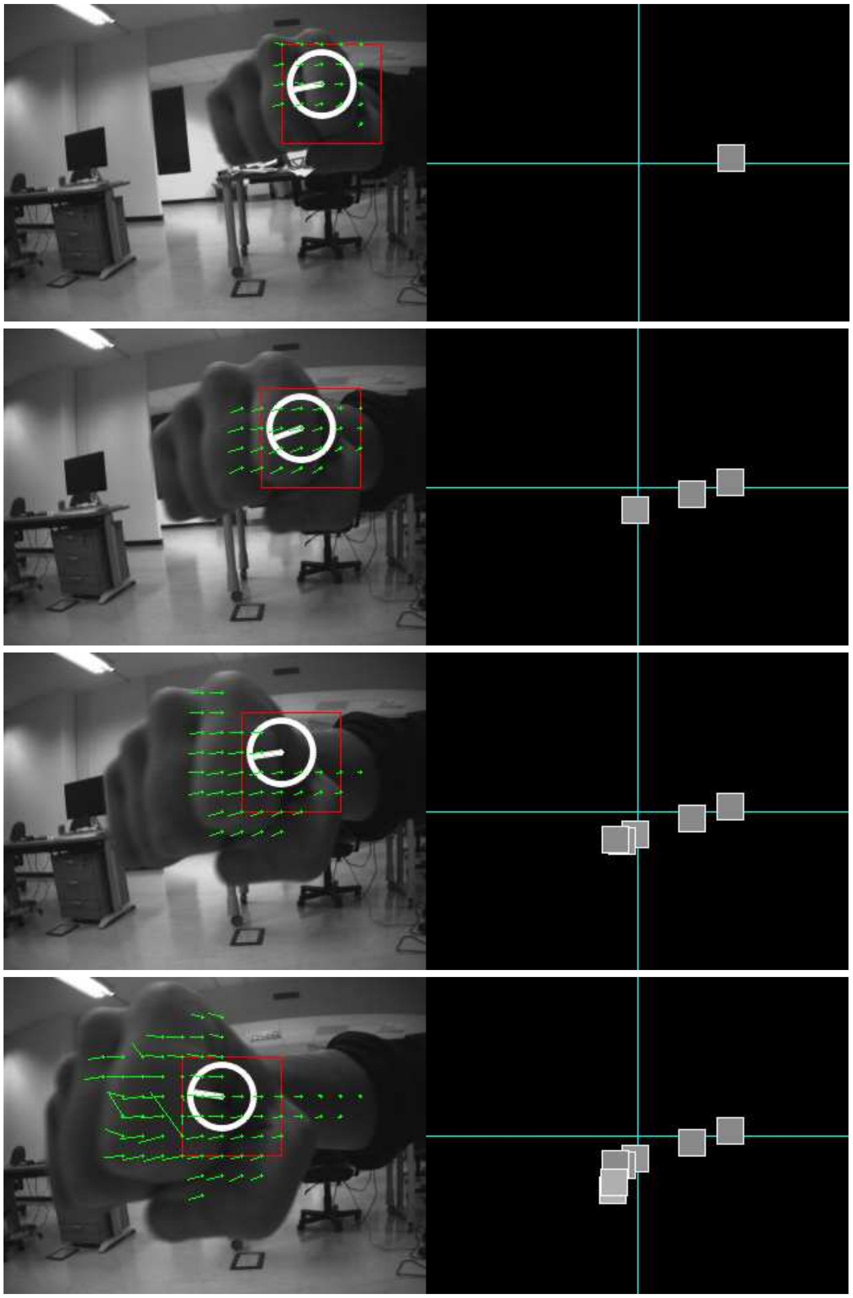

Figure 8.8: Sample output frames from live impact prediction test for an incoming closed fist. Left image shows the optical flow in the segmented region of motion, and the estimated direction of translational motion in the plane. The red box represents the region of maximum divergence. Right image shows estimated location of impact about the camera. Increasing brightness in impact markers represents the estimated time-to-contact. Complete output from the impact prediction algorithm is given in Video 8.1 of the supplementary CD-ROM to this thesis.

al. [2004] suggest this bias may be an evolved defensive adaptation to protect near peri-personal space in humans.

Impact locations are expressed in temporal units, scaled by the time-to-contact of the incoming object with respect to the camera. A potential drawback of this representation is that it denies an absolute mapping of incoming objects to a body centred coordinate frame (although the inclusion of depth information would resolve this). However, an absolute coordinate frame is not necessary to support actions to either avoid, or intercept an incoming object. For example, the act of intercepting an incoming object may be achieved by moving the camera to minimise the estimated temporal distance of the object's predicted impact location with respect to the camera (assuming the camera is the desired point of interception). As a by-product of this action, better estimates of the impact location are also likely to be achieved. Actions to avoid impact with an incoming object can be achieved simply by moving in an opposing direction to the object's estimated trajectory. The level of urgency for such an action is expressed through the time-to-contact itself.

The use of a temporal scale also provides a natural means in which to mimic the velocity-determined variation of the depth of bimodal neuron receptive fields. The initial detection is therefore based on the incoming object's time-to-impact, rather than its crossing of a physically defined threshold. In this way, faster moving objects are detected at a further distance away, as observed in primate experiments [Fogassi et al. 1996].

8.5.1 Future work

The current implementation assumes a stationary camera. It is important to note, however, that this assumption is a choice of convenience to simplify the segmentation of incoming object motion. Subsequent steps of the algorithm are applicable to a translating camera. Future work will consider the use of fast motion segmentation techniques to distinguish regions of self-motion induced flow from regions of flow due to self-motion.

A more biologically plausible model is likely to incorporate learning into the prediction model. Evidence suggests that predictions of impact trajectories are identified in learnt patterns of flow [Ogino et al. 2006]. Localised tactile responses resulting from the impact of visible objects with the body provide instant feedback to a training loop. Patterns of flow in particular regions of the observer's viewing area may then be associated with specific mappings to the body coordinate frame. The incorporation of such learning strategies is beyond the scope of the present study, but will be considered in future work.

The work presented here has considered impact about the infinite camera-centered plane. A significantly more complex problem is how predictions of impact may be obtained for body parts moving independently of the viewing direction. Future work will consider how the proposed scheme may be applied to the more general problem. The underlying neural encoding of trajectory prediction in primates remains an active area of research. The evidence generally supports the view that full trajectory prediction to support object interception tasks is likely to require more than visual cues alone [Zako et al. 2009].

8.6 Summary

Based on neuroscience evidence in primate vision, we have presented preliminary work in developing a scheme for predicting the location of impact for an incoming object, based on the object's instantaneous pattern of optical flow. The scheme estimates the location of impact on a planar body, centred on the location of the camera. We achieve this by examining both the translational component of the object's optical flow, and the divergence. We do not apply tracking, or any learning algorithms to achieve this. Quantitative testing over real image sequences have demonstrated the scheme's ability to achieve a workable accuracy over a range of approach angles. Live testing has also demonstrated the algorithm's real-time use, and its ability to qualitatively distinguish between different approach trajectories. While accuracy appears to degrade for impact points significantly away from the camera, results remain stable and consistent. Future work will consider the problem for a moving camera, and in closed-loop control of a threat avoidance and/or object interception strategy.

In the next chapter we consider the detection of self-moving objects for on-road hazard detection. We consider contact estimation in the context of a moving camera with known motion, and for the detection of non-looming threats (*i.e.,* side-entering).

Chapter 9

On-road hazard detection for driver assistance

9.1 Introduction

In the previous chapter we considered the use of optical flow to estimate the time and location of impact with an incoming self-moving object. However, for a moving observer, potential hazards may also occur when the path of a self-moving object crosses the future path of the observer. Such hazards are unlikely to appear as looming objects, but rather, as objects with a component of motion in the direction of the observer. Thus, divergence-based contact estimation is unlikely to be sufficient for perceiving such hazards.

In this chapter we explore the use of optical flow to detect side-entering hazards. We consider this in the context of a system providing on-road hazard perception assistance for older drivers. This work forms part of a larger collaborative project studying the effects of visual ageing on hazard perception, with an aim towards developing potential interventions to assist older drivers [Horswill et al. 2008]. We present preliminary results of the proposed heuristic-based side-entering hazard detector, tested using real unscripted video footage of potential traffic conflicts as identified by road safety experts. The same footage is to be used in clinical trials with human participants.

The chapter is structured as follows. Section 9.2 discusses background and motivation for this work. Section 9.3 presents the proposed method for detecting side-entering hazards from the optical flow. Section 9.4 provides details of the detectors implementation and describes the methodology of assessment employed. Section 9.5 reports results obtained from the application of the hazard detector over real video footage. Section 9.6 provides a discussion of these results, and future work for the project. Section 9.7 summarises the chapter.

9.2 Background and motivation

There is growing evidence that a driver's ability to perceive hazards declines with age. The likely cause of this is age-related decreases in cognitive and visual functions [Horswill et al. 2008]. Population and case-control studies have found that reaction time, speed of processing, visual selective attention, executive function, eye disease and poor contrast sensitivity are associated with increased crash risk and poorer on-road driving performance [Anstey et al. 2005]. Increased response time for hazard perception in older drivers has been most strongly linked to a loss of contrast sensitivity, and a reduced *useful field of view* [Horswill et al. 2008]. Such visual and cognitive deficits can force older adults to cease driving, despite being otherwise capable. Forced cessation of driving can be especially difficult where public transport is not readily available (particularly rural and outer-suburban communities) and has also been linked to depression in older adults [Marattoli et al. 1997].

A possible alternative is to develop intervening hazard detection technologies that may compensate for the specific visual deficits which cause decreased hazard perception ability. This in turn may allow otherwise capable drivers to keep driving safely, longer.

To this end, the work presented in this section forms part of a collaborative project investigating the effects of cognitive and visual ageing on hazard perception in older drivers[1] . An aim of this project is to pilot possible automated hazard perception interventions that may alert a driver to specific classes of hazards in the scene. Such interventions may then be tested and validated via clinical trials.

[1] Clinical study conducted by the Centre for Mental Health Research, the Australian National University, Canberra and the School of Psychology, University of Queensland, Brisbane.

9.2.1 Previous work

Section 8.2.1 discussed general methods for estimating the motion parameters of self moving objects. As noted, such techniques are currently not feasible for real-time use and thus are not applicable to the work presented here. In the context of road-based hazard detection, constraints on observer and object motion as well as *a priori* knowledge of the camera-vehicle configuration has allowed the inclusion of heuristics to simplify the problem for on-road applications.

A common approach is to apply models of the expected optical flow due to self-motion, thereby identifying regions of the flow field that violate this model. This is often achieved via motion models of the road plane [Braillon et al. 2006; Suzuki and Kanade 1999]. Braillon *et al.* [2006], for example, derive a motion model for the ground plane from odometric information obtained from the vehicle. Using this, they extract the ground plane via a correlation-based generative method. Suzuki and Kanade [1999] apply a parametric estimation model to obtain a camera-mounted vehicle's ego-motion parameters. Song and Chen [2007] propose a system for detecting moving vehicles entering regions on either side of a car. They detect objects that lie on the road plane via feature-based motion estimation and segmentation of flow on the road plane. An issue with road-based motion models is that local intensity variation is often too small to reliably compute optical flow, or extract feature points. In addition, obstacle detection based on the violation of motion models of the road plane ignores the possibility of objects entering from the side.

Previous work in road-based hazard detection typically reports performance results using specific hand-picked scenarios. While advances have been made in the detection of moving obstacles, there has not been any significant study of how such subsystems may actually address the specific needs of a hazard perception assistance systems. Results reported typically do not consider performance over large video sequence sets depicting real, unscripted hazardous scenarios.

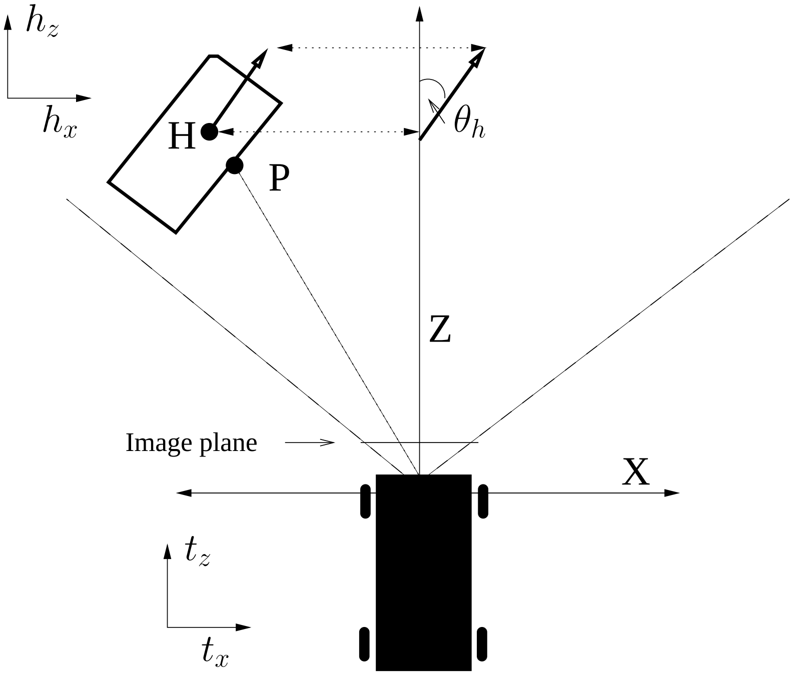

Figure 9.1: Geometric framework for side-entering hazard detection.

9.3 Proposed method

To facilitate a more general framework for hazard detection, we do not incorporate contextual information such as the road plane, or other environmental assumptions. Instead, we focus specifically on the use of early vision cues such as optical flow. Through the estimation and subtraction of optical flow due to self-motion, we identify side-entering hazards with respect to the current direction of motion, from the residual motion due to independently moving objects. Through this, we seek to identify regions of heightened crash risk in the periphery of the image.

We assume a forward facing camera undergoing predominantly translational motion. While we do not apply de-rotation to the flow field, previous chapters have outlined efficient techniques for eliminating rotational flow. We first consider the case of a stationary camera. We then extend this to the case of a moving camera, and present a heuristic-based detection method for identifying side-entering hazards.

9.3.1 Identifying side-entering hazards with a stationary case

Let H be an independently moving object with velocity $\dot{\mathbf{h}} = [h_x\, h_y\, h_z]$. Let θ_h be the direction of motion of H on the ground plane, with respect to the Z axis, such that:

$$\theta_h = \arctan(\frac{h_x}{h_z}). \tag{9.1}$$

We consider H to be a *side-entering* hazard if $h_x < 0$ and $0 \leq \theta_h \leq \frac{\pi}{2}$, or $h_x > 0$ and $-\frac{\pi}{2} \leq \theta_h \leq 0$. That is, H is a side-entering hazard if there exists a component of horizontal motion towards the Z axis. Figure 9.1 shows the geometric framework used.

We seek to infer the direction of motion of H in the $X-Z$ plane from the projection of its apparent motion in the image plane. Let $P = [p_x\, p_y\, p_z]$ be a point on H. Assuming a pinhole camera model with unitary focal length, we project P into the image plane and, considering only translational motion in the $X-Z$ plane, obtain the following equations for the image velocity of P:

$$u_p = \frac{h_z}{p_z}(\frac{h_x}{h_z} - p_x), \tag{9.2}$$

$$v_p = -\frac{p_y h_z}{p_z}, \tag{9.3}$$

where (u_p, v_p) are the horizontal and vertical components of the image velocity.

Notably, u_p and v_p provide a linear system of equations relating the unknown object velocity direction components: h_x and h_z. While obtaining h_x and h_z directly from these equations is not possible, the ratio of these components can be obtained. Rearranging (9.3) such that:

$$\frac{h_z}{p_z} = -\frac{v_p}{p_y}, \tag{9.4}$$

and substituting into (9.2), we obtain:

$$u_p = -\frac{v_p}{p_y}\left(\frac{h_x}{h_z} - p_x\right), \tag{9.5}$$

and after simple algebraic manipulation:

$$\frac{h_x}{h_z} = -\frac{p_y u_p + p_x v_p}{v_p}. \tag{9.6}$$

Thus, we obtain the ratio, $\frac{h_x}{h_z}$ in terms of known and measurable visual quantities. Substituting back into (9.1), we obtain an equation for the direction of motion of H:

$$\theta_h = \tan^{-1}(-\frac{p_y u_p + p_x v_p}{v_p}). \tag{9.7}$$

From this we can identify regions of motion corresponding to side-entering hazards, as defined earlier. Notably, there exists a singularity where $v_p = 0$. This situation, however, can be easily identified, and with added spatial support, should not pose any problems in the computation of θ_h.

9.3.2 Identifying side-entering hazards during forward translation

Let $\dot{\mathbf{t}}_c = [t_x \, t_y \, t_z]$ be the velocity of a camera-mounted vehicle. Assuming translational motion only, the optical flow produced by the motion of the camera is given by:

$$\begin{aligned} u_c &= \frac{t_z}{Z}\Big(\frac{f_x t_x}{t_z} - x\Big), \\ v_c &= \frac{t_z}{Z}\Big(\frac{f_y t_y}{t_z} - y\Big), \end{aligned} \tag{9.8}$$

where f_x and f_y are focal lengths (in pixels), and (x, y) is the projected image location of a point $P = [x \, y \, z]$ in a camera-centred coordinate system. Assuming motion is predominantly on the ground plane, and is approximately along the optical axis, we set t_x and $t_y = 0$.

Consider again an independently moving object, H, observed by the moving camera described above. The optical flow field, $\mathbf{f} = (u, v)$, induced by the movement of H and movement of the camera, is given by the sum of both contributions, such that:

$$\begin{aligned} u &= u_p + u_c \\ &= \frac{1}{Z}\Big(h_x - x(h_z - t_z)\Big) \end{aligned} \tag{9.9}$$

$$\begin{aligned} v &= v_p + v_c \\ &= \frac{1}{Z}\Big(- y(h_z - t_z)\Big) \end{aligned} \tag{9.10}$$

It can be seen from the above that the contribution of h_z is diminished by t_z. We therefore seek to remove the components of flow due to self motion of the camera, such that we obtain Equations 9.2 and 9.3. A true model of self motion, however, requires knowledge of the depth of points in the scene. A common strategy is to estimate self-motion models from the ground plane, where knowledge of the cameras height above the surface may be used to compute a motion model. For the purposes of developing a fast executing first stage hazard detector, however, we do not attempt to compute such models. Rather, we consider only the direction of flow vectors in the image, from which a heuristic-based detector is derived.

Let $\hat{f} = (u', v')$ be the unit vector in the direction of the optical flow vector $\mathbf{f}$, such that:

$$\hat{f} = \frac{\mathbf{f}}{|\mathbf{f}|}, \tag{9.11}$$

where $|\mathbf{f}|$ is the magnitude of the flow vector. Dividing through by this value, we obtain the following directional components for the flow vector:

$$\begin{aligned} u' &= h'_x - x(h'_z - t'_z), \\ v' &= -y(h'_z - t'_z), \end{aligned} \tag{9.12}$$

where h'_x, h'_z and t'_z are the velocity components scaled by $|\mathbf{f}|$.

To approximate the effects of self-motion, we generate an expected pattern of flow directions induced by self-motion in the viewing direction. Considering only the direction, we obtain the simple motion template:

$$u'_c = \frac{x}{\sqrt{x^2+y^2}}, \qquad v'_c = \frac{y}{\sqrt{x^2+y^2}} \tag{9.13}$$

and subtract (u'_c, v'_c) from Equation 9.12 to obtain:

$$u' - u'_c = h'_x - x\Big(h'_z - t'_z + \frac{1}{r}\Big), \tag{9.14}$$

$$v' - v'_c = -y\Big(h'_z - t'_z + \frac{1}{r}\Big), \tag{9.15}$$

where $r = \sqrt{x^2+y^2}$.

Substituting h_z for $h'_z - t'_z + 1$, and (u_p, v_p) for $(u' - u'_c, v' - v'_c)$ in Equation 9.7 we obtain the following equation for the approximation to θ_h $(\tilde{\theta_h})$:

$$\tilde{\theta_h} = \tan^{-1}\Big(\frac{h'_x}{h'_z + \alpha}\Big) = \tan^{-1}\Big(\frac{y(u' - u'_c) + x(v' - v'_c)}{v' - v'_c}\Big) \tag{9.16}$$

where $\alpha = \frac{1}{r} - t'_z$.

As noted, Equation 9.16 provides only an approximation to θ_h. The normalisation of the estimated optical flow field, and the self-motion template, effectively assumes a uniform depth for all points in the scene. It is therefore not possible to correctly account for self-motion in the resulting residual flow. The result of this is an additional contribution to the Z component of the hazard's velocity, represented by α in Equation 9.16. Considering this term, it can be seen that where t'_z is under compensated for (*i.e.*, $t'_z > \frac{1}{r}$), a bias in $\tilde{\theta_h}$ is introduced towards the centre of view. Conversely, where t'_z is over compensated for, a bias towards the direction of observer motion is introduced. We note, however, that for a side-entering hazard to pose an imminent threat of collision, its velocity must be close to, or faster, than observer motion at the point of entry in the visual field. Its apparent motion will therefore be directed towards the centre of view, and always in conflict with self-motion flow directions. Thus, Equation 9.16 will identify all side-entering hazards in the image. Moreover, for threats moving predominantly side-ways, any bias introduced by α is likely to be small with respect to h'_x. Thus, $\tilde{\theta_h}$ should provide a reasonable approximation to θ_h for side-entering hazards posing an immediate threat.

9.4 Implementation of side-entering detector

We implement the side-entering hazard detector by first computing the optical flow field in peripheral regions of the image. These *side-hazard regions* are placed on either side of the estimated location of the focus of expansion (FOE), at a preset horizontal distance, d, from the FOE. For the trials presented here, $d = 0.1 \times \text{image\_width}$. While we assume motion is, on average, in the direction of the optical axis, tracking of the FOE accounts for rotational effects inevitably introduced under real-world conditions. This was also found to be useful for handling uneven motion on the road (*e.g.,* speed bumps). To estimate the location of the FOE, we employ a Hough-based voting [Duda and Hart 1972] approach to find the intersection point of computed optical flow vectors.

9.4.1 Generating the self-motion template

To generate the self-motion template, we also incorporate the FOE location. Directional flow vectors are generated, moving radially away from the estimated location of the FOE. We assume lens distortion is negligible or accounted for through pre-calibration of the camera. In the current implementation, we suppress the output of the detector if the FOE is seen to shift significantly away from the image centre. This template is then subtracted from the estimated unit vector flow field, thus leaving the residual motion defined by Equations 9.14 and 9.15.

9.4.2 Hazard detection

Taking the residual motion field, we convolve a 5×5 weighted window over u_p and v_p separately to obtain the relative support of visual motion in both directions. This is then used to compute $\tilde{\theta_h}$ as defined in Equation 9.16. By considering the computed $\tilde{\theta_h}$ at each image location, side-entering hazard regions are constructed via a simple region-growing technique, whereby neighbouring pixels also classified as side-entering

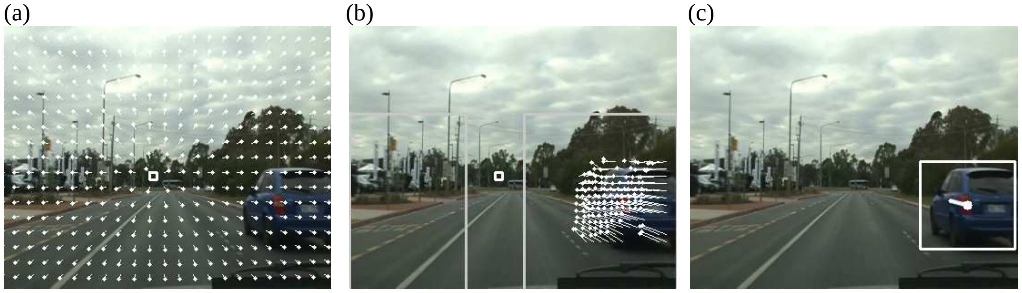

Figure 9.2: A sample side-entering hazard scenario showing from left to right: (a), the computed self-motion template and estimated location of the FOE, (b), the estimated optical flow for the segmented region, and the peripheral regions of interest, and (c), the marked hazard, and estimated direction of motion in the image plane.

hazards are grouped together. From this, a bounding box is computed. Figure 9.2 gives a sample frame showing the estimated FOE, the peripheral regions used for detection, and a region of the image identified as a potential hazard (with optical flow).

To improve robustness to false positives, temporal support is also included. A hazard detection alert is not issued unless the region associated with the possible hazard has been identified as a hazard in the last two updates. If no additional support is received after three frames, the hazard region is considered invalid, and thrown away.

Optical flow is computed using the pyramidal version of Lucas and Kanade's [1981] optical flow method as described in Section 3.2.2.5. To improve the quality of flow estimates, thresholding is applied to the smallest eigenvalue obtained from local covariance matrices computed over 5 × 5 pixel windows. This eliminates regions of low intensity variation from the optical flow computation, where flow is unlikely to be computed accurately. Flow vectors were computed for every 8th pixel, over images of resolution 360 × 288 pixels.

9.4.3 The Hazard Perception Test

The hazard perception test is a video test incorporating local road hazards. The test is an adaptation of a previously used technique developed by Horswill *et al.* [2004]. Participants view video footage of a driver's eye view of various genuine traffic hazards. They are instructed to press a response button when they detect a potential traffic conflict (where the camera car might have to brake or take evasive action to avoid a collision). Reaction times to selected incidents on the video are measured and averaged to give an overall hazard perception reaction score.

The hazard perception test for older drivers was designed in consultation with focus groups and road safety experts who provided information used to identify scenes to be included in the video footage [Horswill and McKenna 2004]. All footage was filmed in normal traffic conditions around areas of the Australian Capital Territory, Australia. Whenever a potential traffic conflict was encountered, its time-code was recorded and indexed. It is important to emphasise that all the clips depict genuine, unstaged hazardous events.

9.5 Performance assessment

9.5.1 Testing procedure

Six video segments from the clinical hazard perception trials were used to assess the accuracy and robustness of the proposed hazard detector. From the set of all indexed hazards across the video segments, those fitting the description of side-entering were marked as hazards to detect. Start and end time-codes listed for each of the indexed side-entering hazards were used to define the duration of time in which the detector must locate the hazard. The detector indicated the existence of a potential side-entering hazard by drawing a bounding box around the image region associated with the potential threat. A hazard was deemed to be detected if the centre of the bounding box hazard region was within the image area of the object causing the hazard.

Table 9.1 provides a full list of all indexed side-entering hazards. A brief description of the side-entering scenario is given, along with the time interval defined for the hazard. Note that the time interval is the same as that used in human trials with the same footage.

9.5.2 Results

The right two columns of Table 9.1 provide results obtained from the application of the hazard detector over the hazard perception test videos. Where the detector successfully

Indexed Hazards			**Detector Results**	
Hazard description	**Time duration**		**Time detected**	**Response time**
	start	end (sec)	(sec)	(sec)
Video segment 1				
merge in front from right	2.96	11.75	2.80	-0.16
merge in front from right	21.60	32.00	24.39	+2.79
car turns out from left	49.72	55.96	51.91	+2.19
pedestrians crossing road	140.84	157.36	153.85	+13.01
bus pulls out	161.839	178.05	178.32	+16.92
truck merges from left	251.04	283.60	271.45	+20.41
Video segment 2				
pedestrian crossing from right	8.95	20.20	15.72	+6.77
van swerves right from left	56.24	64.94	57.47	+1.23
pedestrian crossing from right	145.83	156.76	–	–
car turns out from left	202.80	209.16	205.08	+2.28
bus turns out from left	253.16	264.14	255.21	+2.05
Video segment 3				
car merges right	22.91	39.65	16.67	-6.24
pedestrians crossing from left	130.82	141.88	–	–
pedestrian crossing from right	240.86	245.52	243.59	+2.73
Video segment 4				
truck pulling out from right	0	13.08	9.25	+9.25
car turns out from left	75.64	83.24	80.17	+4.53
car merges from left	153.793	169.24	152.92	-0.87
car on round-about from right	170.28	179.36	172.18	+1.9
pedestrian moving from left	261.73	274.04	–	–
Video segment 5				
car merges from left	23.79	39.08	26.64	+2.85
pedestrian crossing from right	54.88	63.61	–	–
pedestrians crossing from left	98.69	105.93	101.78	+3.09
car crosses road from right	107.32	114.60	109.84	+2.52
pedestrian crossing from right	157.32	165.13	162.20	+4.88
pedestrian crossing from right	204.20	215.6	–	–
Video segment 6				
car turns out from right	11.32	19.18	13.40	+2.08
bus starts pulling out from left	65.49	77.20	75.21	+9.72
pedestrians crossing from right	84.05	96.30	91.36	+7.31
van pulls out from left	126.16	139.68	138.46	+12.3
bus pulls out from left	152.84	174.5	166.93	+14.09
truck enters round-a-bout from left	186.72	226.39	–	–

Table 9.1: Results obtained using the side-entering hazard detector across all video segments used in hazard perception testing of older drivers. The first three columns provide a description and time interval of all indexed hazards fitting the description of side-entering. The right two columns provide results for the detector in identifying each indexed hazard. Where successful, both the time-code and the response time with respect to the indexed time-code are given.

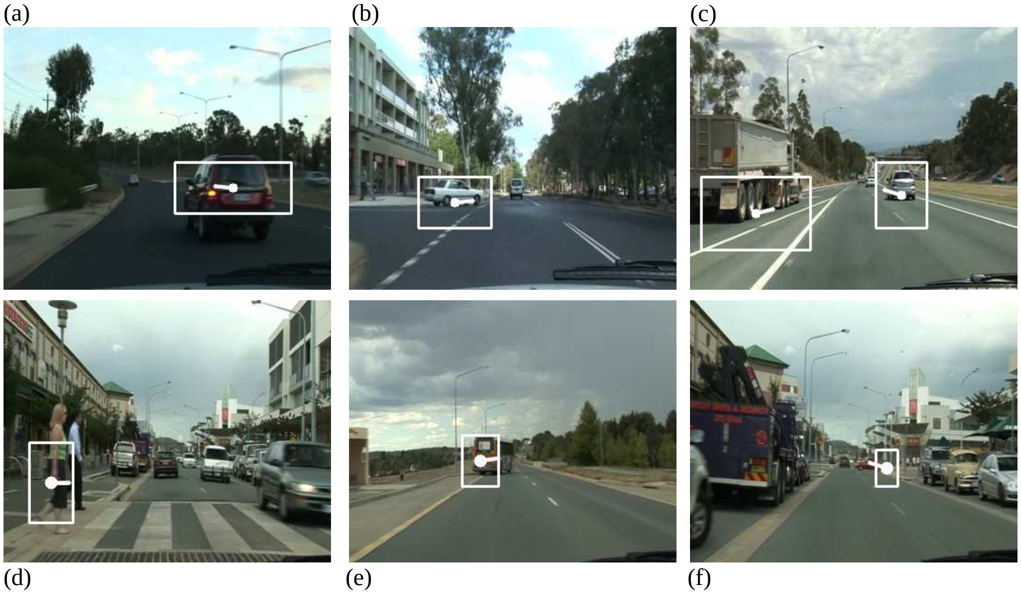

Figure 9.3: Sample hazard detections from video segment (vs) testing: (a) merge from right (vs 1, 29.36 sec), (b) car turns out from left (vs 1, 52.79 sec), (c) truck merges from left (vs 1, 271.45 sec), (d) pedestrians crossing (vs 5, 101.78 sec), (e) bus pulls out from left (vs 6, 166.93 sec), (f) car crosses road from right (vs 5, 109.84 sec)

identified the hazard, both the time-code of the initial detection, and the time difference of this detection with respect to the indexed time-code are given. Figure 9.3 shows a sample of *side-entering* hazard detections recorded over the video segments. Each corresponds to a successfully detected indexed hazard in Table 9.1, as indicated by the video segment number and time-code listed with each sample.

Across the six video segments, the hazard detector successfully identified 24 of the 30 (80%) indexed side-entering hazards. In addition, the detector was observed to detect a significant number of other side-entering scenarios, not marked for detection.

Hazard class	Detections	Avg response time (secs)
Vehicle side road entry	7/8	+2.27
Vehicle merge (or swerve) to front	6/6	+3.4
Pedestrian(s) crossing	6/11	+6.3
Vehicle pull out from curb	5/5	+12.47
Total	24/30	+5.65

Table 9.2: Hazard detection results broken down to major side-entering hazard classes.

Of the total number of hazard alerts issued, 41% were observed to be false positives. A false positive was deemed to be any hazard alert not involving a self-moving object (in practise a moving vehicle or person).

9.6 Discussion

Overall, results from the hazard perception trial are encouraging. Table 9.2 shows a breakdown of performance statistics into the major classes of side-entering hazards. The strongest results achieved for speed of detection involve situations where a vehicle enters the field of view moving. This is in contrast to the worst performing scenario for detection time involving vehicle's pulling out from an initially stationary position. It should be noted, however, that the indexed start time of these hazards is significantly earlier than when the vehicle starts to move. Arguably, these hazards constitute a *stopped vehicle in lane* hazard rather than side-entering when they initially enter view. Notably, however, all such hazards were detected within the allotted time.

Pedestrian-related hazards posed the greatest challenge where only 6 from 11 were identified. This is in contrast to the strong results achieved for side-entering vehicles, a result that is also reflected in average response times. Successful pedestrian-related detections took, on average, twice as long as side-entering vehicle detections. The likely cause of this discrepancy is the relatively slow apparent motion, and small size of pedestrians as compared with vehicles. Pedestrians were generally only detected once the camera-mounted vehicle came to a halt. Another observed difficulty in detecting pedestrians was that in many cases, pedestrians entered the field of view as stationary objects, often waiting to cross the road. This suggests motion-based cues are unsuitable for early pedestrian detections. Rather, specifically designed pedestrian classification techniques such as that outlined by Overett *et al.* [2009], are more appropriate for this class of detection.

While the number of recorded false positives is significant, their occurrence was predictable, and limited to specific environmental scenarios. Of the total number of false positives recorded, 83% were found to be the result of lines, shadows, and other features on the surface of the road. In many cases, these features would remain for a significant time, thus causing repeated detections. Given an extraction of the road-plane, such false detections should easily be filtered out, thus reducing false-positives to 7%. Other false positive detections were typically associated with cars in the distance, driving toward the camera on a curved section of road. Such scenarios were often observed to generate almost identical apparent motion to side-entering hazards.

In general, the use of low-level visual cues like flow direction will always be susceptible to noise and ambiguities in the scene. It is therefore unlikely that such a system would be applied without higher-level contextual information. As stated in the introduction, the goal of the current system is to identify image regions of heightened threat risk. The inclusion of other visual cues would further improve the robustness and effectiveness of the system.

9.6.1 Future work

Future work will consider other classes of hazards, and the use of other visual information to detect hazards. Cues such as flow field magnitude and divergence (or looming) provide a direct measure of the relative proximity of objects in the scene. Such cues may also provide a means of gauging the level of threat posed by environmental conditions in general. For example, increasing flow magnitude in the periphery would suggest conditions are narrowing, thus increasing the risk of pedestrians or other objects entering from the side. Such a cue may be used to lower thresholds for the detection of such hazards, and adjust the size of search regions in the image. To facilitate more pre-emptive hazard perception, the inclusion of subsystems to detect more contextual cues such as road signs, flashing indicators and stop lights could be considered.

9.7 Summary

In this chapter we have explored visual contact estimation for non-looming objects and a moving observer. We have reported on preliminary work towards the development of a potential hazard perception intervention to assist older drivers. A class of hazard identified as a cause of heightened crash risk are those involving side-entering objects entering the field of view in the periphery. We have proposed a simple and efficient strategy for identifying regions of the image where the likelihood of such hazard is high. Unlike previous approaches, we assess the performance and effectiveness of the detector over six video segments also being used in concurrent clinical trials of older drivers. In this study, we aim to adapt what is learnt through clinical trials, and pilot an intervention to improve hazard perception in older drivers. Such a system may allow older drivers to keep driving safely, for longer.

We have now presented all contributions of this thesis. In the next chapter we present the conclusions of this thesis, and summarise the major contributions and implications of the work presented.

Chapter 10

Conclusion and Future Work

This thesis has proposed new visual cues for estimating contact with surfaces in the environment. We have focussed specifically on techniques for extracting visual information from the optical flow field, as motivated by ecological studies of biological vision systems. As a primary focus, we have considered visual contact estimation for the purposes of vision guided robot navigation. In particular, we have proposed novel visuo-motor control schemes suitable for tasks requiring fine motion control in close proximity with looming surfaces such as landing and docking. We have also applied directly available cues from the optical flow field to perceptual tasks such as on-road hazard detection and egocentric time and location of impact prediction. To contrast with these direct perceptual techniques, we have also explored visual contact estimation under a more traditional structure-from-motion framework. In so doing we have examined the implications of employing a spherical projection model (as an approximation to the eye geometry of flying insects) for structure-from-motion recovery, as well as for visual contact estimation using directly perceived visual cues such as time-to-contact. Through this work, we have gained new insights into the role of vision in navigation and perception, and how vision algorithms may best support the needs of visuo-motor control. In this chapter, we summarise these insights, and the novel contributions of this thesis. We also discuss future directions for this research.

10.1 Key insights for visual contact estimation

Visual contact estimation under imprecise conditions

Global invariants of the optical flow field express relationships between the moving observer and its environment. Understanding these relationships, and extracting visual information with respect to changes of such invariants allows system dynamics to be handled in the image domain. This alleviates restrictions on observer motion, and reduces the demands on subsequent processes to filter visual inputs. This thesis has demonstrated that such an approach may lead to simpler, more robust and generally applicable visual control schemes.

Projective geometries

The choice of camera projection model determines how visual information is expressed, and as a consequence, how it is extracted. The results presented in this thesis support the view that a spherical projection model over a wide field of view is well suited to flow-based navigation and surface contact estimation. Moreover, this thesis has shown that this projection model provides an inherent advantage for estimating time-to-contact and applying it to visuo-motor control schemes. The geometric equivalence of image points under this model alleviates assumptions and restrictions imposed under a perspective projection model when estimating time-to-contact. Results also support previously noted geometric advantages for recovering real-time structure-from-motion from spherical optical flow. However, inherent issues associated with structure-from-motion recovery remain evident, and problematic for use in the control loop.

Qualitative cues for visual control

Robust visual control does not require explicit estimation of egomotion and structure parameters. Moreover, the results presented in this thesis support the view that visual control systems that avoid a direct need for egomotion estimates (or an assumed direction of motion) may perform better when visual conditions are changing rapidly. In this context, direct visual cues reflecting frame-to-frame changes in the relationship between observer motion and scene structure provide a reliable and stable control input. This reduces potential sources of measurement error introduced by egomotion estimation. Under real-world conditions, such errors are difficult to model, and thus necessitate complex control schemes to account for changing conditions.

10.2 Core contributions of this thesis

This thesis has contributed the following:

A robust strategy for docking a mobile robot using optical flow field divergence

We have proposed a mobile robot docking strategy that utilises a time-to-contact estimation that is robust to noisy, instantaneous rotations induced by robot ego-motion. We have shown that through tracking the focus of expansion in the optical flow field, small rotations of the camera and misalignments of the optical and translational axes can be accounted for by calculating flow divergence with respect to the FOE. Based on this, we have proposed a divergence-based control law for docking a robot with near fronto-parallel surfaces, verified through experimental trials. The accuracy and stability achieved is demonstrated to be sufficient for fine motion control of a mobile robot when in close proximity with an upright surface.

A unified strategy for landing and docking from spherical flow divergence

We have proposed a unified landing and docking strategy using flow field divergence under spherical projection. We have derived a divergence-based velocity and direction control strategy that can be applied for any angle of approach, and without restriction on the motion of the camera (or the need for explicit de-rotation of the flow field, or egomotion estimation). Central to this strategy is the use of the point of maximum divergence. Analytically, and experimentally, we have shown that the divergence maximum always occurs halfway along the arc connecting the surface normal and the direction of translation. We have demonstrated that such a strategy is viable for closed-loop use, and provides robust estimates of time-to-contact under real-world conditions.

A real-time strategy for estimating 3D depthmaps from spherical flow

We have presented an insect-inspired structure-from-motion strategy for generating 3D relative depth maps from optical flow, in real-time. In so doing, we have demonstrated the advantages of a spherical projection model over a full view sphere (or hemisphere for planar motion) when recovering egomotion parameters. We have shown for the first time, the application of an egomotion algorithm first proposed by Nelson and Aloimonos [1988], over real images, depicting real world environments. Results suggest this strategy may be a useful base for high-level navigation tasks.

A time and location of impact estimator for incoming self-moving objects

Based on primate vision, we have presented work towards the development of a flow-based scheme for predicting the impact location of an incoming object about an extended camera-centred plane. The proposed method bases predictions on the examination of local instantaneous flow induced by the motion of an incoming object. We have presented both off-line quantitative testing, as well as live testing of the system. Results provide encouragement for future work.

A flow-based hazard alert system for classes of on-road hazards

We have presented preliminary work towards the development of an on-road hazard alert system to compensate for visual ageing in older drivers. We have proposed a simple and efficient strategy for identifying regions of the image where a heightened risk of collision exists. We have focussed on the detection of peripheral side-entering hazards, which have been identified to be particularly problematic. Unlike previous approaches, we have assessed the performance of the detector over video segments used in concurrent clinical trials of older drivers. Results showed the detector successfully identified almost all of the indexed hazards.

10.3 Further work

Here we discuss future research directions and extensions in the estimation and application of visual contact estimation. Note that future work for some contributions of this thesis have already been discussed in the corresponding chapter, and are not repeated here.

Extended uses of divergence maxima

In Chapter 6 we demonstrated the use of the divergence maximum within the projected area of a planar surface for docking and landing. The exploration of extended uses and alternative applications offers potential direction for further work. Given a global field of view, for example, multiple local divergence maxima (and minima) are likely to be present in the image, each associated with different surfaces. Such cues offer potential new approaches to visuo-motor navigation in structured environments. Examination of divergence within the neighbourhood of local maxima may also provide a useful cue for local plane fitting using models of planar divergence. Divergence maxima and minima over a wide field of view may also provide a useful constraint for egomotion recovery under certain visual conditions.

Optical flow estimation

This thesis has applied a single class of optical flow estimation techniques in all reported experiments. While this choice is well supported by previous benchmarking and comparison studies (discussed in Chapter 3), there remains scope for further consideration of which technique best serves the needs of visual contact estimation. Discrete feature matching techniques such as SIFT may also be used in place of differential optical flow estimation. While computationally expensive to compute, sparse feature-sets may provide sufficient information to infer visual information such as divergence across surfaces.

Alternative sensory inputs

This thesis has considered visual contact estimation using optical flow. However, performance in sparsely textured, featureless environments, or where lighting conditions may significantly change, represent conditions in which optical flow (and feature-matching) are unlikely to perform well. Visual cues such as surface shading, object boundaries (via edge detection), and closed-shape tracking provide alternative cues for perceiving the changing proximity and orientation of surfaces in the scene. Future work may seek to employ such cues separately, or in conjunction with the flow-based techniques proposed in this thesis.

Extended closed-loop verification

More extensive closed-loop trials of all proposed techniques are required. In particular, closed-loop verification of the depth map estimation scheme, and time and location of impact detector are yet to be conducted. Across all visuo-motor control schemes proposed, closed-loop performance in real-world conditions remains untested. While open-loop experimental results have demonstrated robustness in the underlying control input signal, demonstration of closed-loop performance under such conditions would provide confirmation of this. Restrictions imposed by the robotic platforms used in this thesis disallowed closed-loop outdoor experiments to be conducted.

Embedding visual contact estimation in visual behaviours

Several techniques proposed in this thesis are yet to be applied to specific visual behaviours, and this remains future work. More generally, this thesis has not considered how any of the proposed contact estimation schemes may be embedded in goal driven navigation and perceptual systems. A possible direction of future research is to consider how visuo-motor control schemes may be combined with structure-from-motion techniques to support a broad array of navigation and perceptual capabilities.

Biologically-inspired versus conventional structure-from-motion

This thesis emphasises the importance of biological approaches to visual navigation. We acknowledge, however, that alternative solutions exist, and in many cases, may provide superior performance. Classical structure-from-motion approaches are beginning to realise real-time performance, and provide increasingly detailed scene reconstructions. It will be of interest to compare such approaches with those presented in this thesis (and elsewhere) in the control loop. How classical structure-from-motion may be integrated with low-level visuo-motor schemes is also a topic for future work.

Biological implications

Much of the work presented in this thesis has been broadly based on principles of vision in biology. While the focus has been on the design of robust vision algorithms for real-time, real-world applications, outcomes of this work may offer potential insights into the visuo-motor control capabilities in animals. Of particular interest is the proposed unification of docking and landing, which offers an alternative and more general model for graze-landing honeybees than currently exists in the literature. From a broader perspective, further research may consider the possible role of the divergence maxima for other visuo-motor animal behaviours, including fixation-based primate vision for which rotationally invariant cues are particularly useful. Development of the time and location of impact prediction scheme presented in Chapter 8 will also provide a base for modelling visuo-motor strategies for threat avoidance and interception in primates.

Bibliography

Adiv, G. 1985. Determining three-dimensional motion and structure from optical flow generated by several moving objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence 7*, 4, 384–401.

Adiv, G. 1989. Inherent ambiguities in recovering 3-d motion and structure from a noisy flow field. *IEEE Transactions on Pattern Analysis and Machine Intelligence 11*, 5, 477–489.

Akbarzadeh, A., Frahm, J.-M., Mordohai, P., Clipp, B., Engels, C., Gallup, D., Merrell, P., Phelps, M., Sinha, S., talton, B., Wang, L., Yang, Q., Stewenius, H., yang, R., Welch, G., Towles, H., Nistèr, D., and Pollefeys, M. 2006. Towards urban 3d reconstruction from video. In *Proceedings of the Third International Symposium on 3D Data Processing, Visualization and Transmission (3DPVT)* (2006).

Aloimonos, J., Weiss, I., and Bandopadhay, A. 1987. Active vision. *International Journal of Computer Vision 1*, 4, 333–56.

Aloimonos, J. Y. 1993a. *Active Perception.* Lawrence Erlbaum Associates, Publishers.

Aloimonos, J. Y. 1993b. *Introduction: Active Vision Revisited*, pp. 1–18. Lawrence Erlbaum Associates, Publishers.

Alvarez, L., Weickert, J., and Sánchez, J. 1999. Reliable estimation of dense optical flow fields with large displacements. *International Journal of Computer Vision 39*, 1, 41–56.

Ancona, N. and Poggio, T. 1993. Optical flow from 1d correlation: Application to a simple time-to-crash detector. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (1993), pp. 209–14.

ANSTEY, K., WOOD, J., LORD, S., AND WALKER, J. 2005. Cognitive, sensory and physical factors enabling driving safety in older adults. *Clinical Psychology Review 25*, 45–65.

ARGYRIOU, V. AND VLACHOS, T. 2006. A study of sub-pixel motion estimation using phase correlation. In *Proceedings of the British Machine Vision Conference (BMVC)*, Volume 1 (2006), pp. 387–396.

ARIS, R. 1962. *Vectors, tensors and the basic equations of fluid mechanics.* Prentice-Hall, Englewood Cliffs, N.J.

ARKIN, R. 1998. *Behaviour-Based Robotics.* MIT Press, Cambridge MA, USA.

ARKIN, R. AND MURPHY, R. 1990. Autonomous navigation in a manufacturing environment. *Robotics and Automation, IEEE Transactions on 6*, 4 (Aug), 445–454.

ARMINGOL, J. M., DE LA ESCALERA, A., HILARIO, C., COLLADO, J. M., CARRASCO, J. P., FLORES, M. J., PASTOR, J. M., AND RODRÍGUEZ, F. J. 2007. IVVI: Intelligent vehicle based on visual information. *Robotics and Autonomous Systems 55*, 12, 904–916.

ARNSPANG, J., HENRIKSEN, K., AND STAHR, R. 1995. Estimating time to contact with curves, avoiding calibration and aperture problem. In *Proceedings of CAPI'95 (Computer Analysis of Images and Patterns)* (1995), pp. 856–861.

AZINHEIRA, J. R., RIVES, P., CARVALHO, J. R. H., SILVEIRA, G. F., DE PAIVA, E. C., AND BUENO, S. S. 2002. Visual servo control for the hovering of an outdoor robotics airship. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)* (2002), pp. 2787–92.

BAKER, S., SCHARSTEIN, D., ROTH, J. P. L. S., BLACK, M. J., AND SZELISKI, R. 2007. A database and evaluation methodology for optical flow. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (2007), pp. 1–8.

BALLARD, D. H. 1991. Animate vision. *Artificial Intelligence 48*, 1, 57–86.

BARNES, N. AND LIU, Z.-Q. 2004. Embodied categorisation for vision-guided mobile robots. *Pattern Recognition Letters 37*, 2, 299–312.

Barnes, N. and Sandini, G. 2000. Direction control for an active docking behaviour based on the rotational component of log-polar optic flow. In *Proceedings of the 2000 European Conference on Computer Vision (ECCV). Lecture Notes in Computer Science, Computer Vision* (2000), pp. 167–81.

Barron, J. L., Fleet, D. J., and Beauchemin, S. S. 1994. Performance of optical flow techniques. *International Journal of Computer Vision 12*, 1, 43–77.

Beauchemin, S. S. and Barron, J. L. 1995. The computation of optical flow. *ACM Computing Surveys 27*, 3, 433–67.

Bermùdez, S., Pyk, P., and Verschure, P. F. M. J. 2007. A fly-locust based neuronal control system applied to an unmanned aerial vehicle: the invertebrate neuronal principles for course stabilization, altitude control and collision avoidance. *The International Journal of Robotics Research 26*, 7, 759–772.

Birk, A. and Carpin, S. 2006. Rescue robotics - a crucial milestone on the road to autonomous systems. *Autonomous Systems 20*, 5, 595–605.

Bolles, R. C., Baker, H. H., and Marimont, D. H. 1987. Epipolar-plane image analysis: An approach to determining structure from motion. *International Journal of Computer Vision 1*, 1, 7–55.

Borst, A. and Egelhaaf, M. 1993. Detecting visual motion: Theory and models. In F. A. Miles and J. Wallman Eds., *Visual Motion and its Role in the Stabilization of Gaze*, pp. 3–27. Elsevier Science.

Bouguet, J.-Y. 2000. Pyramidal implementation of the Lucas Kanade feature tracker description of the algorithm. In *OpenCV Documentation*. Intel Corporation.

Bower, T. G. R. and Broughton, J. M. 1970. The coordination of visual and tactual input in infants. *Perception and Psychophysics 8*, 51–3.

Braillon, C., Pradalier, C., Crawley, J., and Laugier, C. 2006. Real-time moving obstacle detection using optical flow models. In *Proceedings of the 2006 IEEE Intelligent Vehicles Symposium* (2006), pp. 466–71.

Branca, A., Stella, E., and Distante, A. 2000. Passive navigation using egomotion estimates. *Image and Vision Computing 18*, 10, 833–841.

Brodsky, T., Fermüller, C., and Aloimonos, Y. 1998. Direction of motion fields are hardly ever ambiguous. *International Journal of Computer Vision 26*, 1, 5–24.

Brooks., R. 1986. A robust layered control system for a mobile robot. *IEEE Transactions on Robotics and Automation 2*, 1, 14–23.

Brooks, R. 1990. Elephants don't play chess. In P. Maes Ed., *Designing Autonomous Agents*, pp. 3–15. Cambridge MA, USA: MIT Press.

Brooks, R. and Stein, L. 1994. Building brains for bodies. *Autonomous Robots 1*, 1, 7–25.

Brox, T., Bruhn, A., Papengerg, N., and Weickert, J. 2004. High accuracy optical flow estimation based on a theory of warping. In *Proceedings of the 2004 European Conference on Computer Vision (ECCV). Lecture Notes in Computer Science, Computer Vision*, Volume 4 (2004), pp. 25–36.

Bruhn, A., Weickert, J., and Schnörr, C. 2005. Lucas/Kanade meets Horn/Schunck: Combining local and global optic flow methods. *International Journal of Computer Vision 61*, 3, 211–231.

Bruss, A. R. and Horn, B. K. P. 1983. Passive navigation. *Computer Vision, Graphics, and Image Processing 17*, 1, 3–20.

Burschka, D., Lee, S., and Hager, G. 2002. Stereo-based obstacle avoidance in indoor environments with active sensor re-calibration. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Volume 2 (2002), pp. 2066–2072.

Camus, T. 1994. *Real-Time Optical Flow.* PhD thesis, Department of Computer Science, Brown University, Providence, USA.

Camus, T. 1997. Real-time quantized optical flow. *Real-Time Imaging 3*, 2, 71–86.

Camus, T., Coombs, D., Herman, M., and Hong, T.-H. 1996. Real-time single-workstation obstacle avoidance using only wide-field flow divergence. In *Proceedings of the International Conference on Pattern Recognition (ICPR)* (1996), pp. 323–30.

Caraffi, C., Cattani, S., and Grisleri, P. 2007. Off-road path and obstacle detection using decision networks and stereo vision. *IEEE Transactions on Intelligent Transportation Systems 8*, 4, 607–618.

Chahl, J. S. and Srinivasan, M. V. 1997. Range estimation with a panoramic visual sensor. *Journal of the Optics Society of America - A 14*, 9 (sep), 2144–50.

Chahl, J. S., Srinivasan, M. V., and Zhang, S. W. 2004. Landing strategies in honeybees and applications to uninhabited airborne vehicles. *International Journal of Robotics Research 23*, 2, 101–110.

Chen, Z. and Birchfield, S. T. 2006. Qualitative vision-based mobile robot navigation. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)* (May 2006), pp. 2686–2692.

Cheng, G. and Zelinsky, A. 1998. Goal-oriented behaviour-based visual navigation. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Volume 4 (May 1998), pp. 3431–3436.

Chiuso, A., Brockett, R., and Soatto, S. 2000. Optimal structure from motion: Local ambiguities and global estimates. *International Journal of Computer Vision 39*, 3, 95–228.

Chiuso, A., Favaro, P., Jin, H., and Soatto, S. 2000. 3-d motion and structure from 2-d motion causally integrated over time: implementation. In *Proceedings of the 2000 European Conference on Computer Vision (ECCV). Lecture Notes in Computer Science, Computer Vision*, Volume 1842 (2000), pp. 735–750.

Cipolla, R. and Blake, A. 1997. Image divergence and deformation from closed curves. *International Journal of Robotics Research 16*, 1, 77–96.

Cohen, I. and Medioni, G. 1998. Detecting and tracking moving objects in video from an airborne observer. In *Proceedings of IEEE Image Understanding Workshop* (1998), pp. 217–22.

Cole, L. and Barnes, N. 2008. Insect-inspired three dimensional centring. In *Proceedings of the Australiasian Conference on Robotics and Automation (ACRA)* (2008).

Colombo, C. 2000. Time to collision from first-order spherical image motion. *Robotics and Autnomous Systems 31*, 5–15.

Colombo, C. and Del Bimbo, A. 1999. Generalized bounds for time to collision from first-order image motion. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Volume 1 (1999), pp. 220–226.

Coombs, D., Herman, M., Hong, T., and Nashman, M. 1998. Real-time obstacle avoidance using central flow divergence, and peripheral flow. *IEEE Transactions on Robotics and Automation 14*, 1, 49–59.

Coombs, D. and Roberts, K. 1993. Centering behaviour using peripheral vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Los Alamitos, CA, USA, 1993), pp. 440–5. IEEE Comput. Soc. Press.

Correa, F. and Okamoto, J. 2005. Omnidirectional stereovision system for occupancy grid. In *Proceedings of the International Conference on Advanced Robotics (ICAR '05)* (2005), pp. 628–634.

Csorba, M. 1997. *Simultaneous Localisation and Map Building.* PhD thesis, University of Oxford, Oxford, UK.

Dailey, M. and Parnichkun, M. 2006. Simultaneous localization and mapping with stereo vision. In *Proceedings of International Conference on Control, Automation, Robotics and Vision* (2006).

Daniilidis, K. and Nagel, H.-H. 1993. The coupling of rotation and translation in motion estimation of planar surfaces. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (1993), pp. 188–193.

Dao, N. X., You, B.-J., and Oh, S.-R. 2005. Visual navigation for indoor mobile robots using a single camera. In *Proceedings of the IEEE/RSJ International Conference on Robots and Intelligent Systems (IROS)* (2005), pp. 1992–1997.

Davison, A., Reid, I., and Molton, N. 2007. MonoSLAM: Real-time single camera SLAM. *IEEE Transactions on Pattern Analysis and Machine Intelligence 29*, 6, 1052–1067.

Dawkins, R. 1996. *Climbing Mount Improbable.* Norton, New York.

Dengate, R., barnes, N., Lim, J., Luu, C., and Guymer, R. 2008. Real time motion recovery using a hemispherical sensor. In *Proceedings of the Australiasian Conference on Robotics and Automation (ACRA)* (2008).

Dissanayake, M. W. M. G., Newman, P., Clark, S., Durrant-Whyte, H. F., and Csorba, M. 2001. A solution to the simultaneous localization and map building (slam) problem. *IEEE Transactions on Robotics and Automation 17*, 3 (Jun), 229–241.

Duchon, A. P. and Warren, W. H. 1994. Robot navigation from a gibsonian viewpoint. In *Proceedings of the 1994 IEEE International Conference on Systems, Man and Cybernetics. San Antonio* (1994), pp. 2272–7.

Duda, R. O. and Hart, P. E. 1972. Use of the hough transformation to detect lines and curves in pictures. *Commications of the ACM 15*, 1115.

Duffy, C. J. and Wurtz, R. H. 1997. Medial superior temporal area neurons respond to speed patterns in optic flow. *The journal of Neuroscience 17*, 8, 2839–51.

Duric, Z., Rosenfeld, A., and Duncan, J. 1999. The applicability of green's theorem to computation of rate of approach. *International Journal of Computer Vision 31*, 1, 83–98.

Durrant-Whyte, H. 1988. Uncertain geometry in robotics. *IEEE Transactions on Robotics and Automation 4*, 1, 23–31.

Durrant-Whyte, H. and Bailey, T. 2006. Simultaneous localization and mapping (SLAM): part i. *IEEE Robotics and Automation Magazine 13*, 2, 99–110.

Elinas, P., Sim, R., and Little, J. J. 2006. δ SLAM: Stereo vision SLAM using the rao-blackwellised particle filter and a novel mixture proosal distribution. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)* (2006).

Fermüller, C. and Aloimonos, Y. 1998. Ambiguity in structure from motion: sphere versus plane. *International Journal of Computer Vision 28*, 2, 137–54.

Fermüller, C. and Aloimonos, Y. 2000. Geometry of eye design: Biology and technology. In T. H. R. Klette and G. Gimelfarb Eds., *Multi Image Search and*

Analysis, Lecture Notes in Computer Science. Springer Verlag, Heidelberg.

Fermüller, C., Shulman, D., and Aloimonos, Y. 2001. The statistics of optical flow. *Computer Vision and Image Understanding: CVIU 82*, 1, 1–32.

Fischler, M. A. and Bolles, R. C. 1981. Random sample consensus: A paradigm for model fitting with application to image analysis and automated cartography. *Communications of the ACM 24(6)*, 381–95.

Fleet, D. J. and Jepson, A. D. 1990. Computation of component image velocity from local phase information. *International Journal of Computer Vision 5*, 1, 77–104.

Fogassi, L., Gallese, V., Fadiga, L., Luppino, G., Matelli, M., and Rizzolatti, G. 1996. Coding of peripersonal space in inferior premotor cortex (area f4). *Journal of Neurophysiology 76*, 1 (July), 141–157.

Gaussier, P., Joulain, C., Zrehen, S., Banquet, J., and Revel, A. 1997. Visual navigation in an open environment without map. In *Proceedings of the IEEE/RSJ International Conference on Robots and Intelligent Systems (IROS)*, Volume 2 (1997), pp. 545–550 vol.2.

Gee, A., Chekhlov, D., Calway, A., and Mayol-Cuevas, W. 2008. Discovering higher level structure in visual SLAM. *IEEE Transactions on Robotics 24*, 5, 980–990.

Gibson, J. J. 1950. *The Perception of the Visual World*. Houghton Mifflin, Boston, MA, USA.

Gibson, J. J. 1966. *The Senses Considered as Perceptual Systems*. Houghton Mifflin, Boston, MA, USA.

Gibson, J. J. 1979. *The Ecological Approach to Visual Perception*. Houghton Mifflin, Boston, MA, USA.

Giralt, G., Sobek, R., and Chatila, R. 1979. A multi-level planning and navigation system for a mobile robot; a first approach to hilare. In *Proceedings of the Sixth International Joint Conference on Artificial Intelligence*, Volume 1 (1979), pp. 335–337.

Green, W., Oh, P., and Barrows, G. 2004. Flying insect inspired vision for autonomous aerial robot maneuvers in near-earth environments. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Volume 3 (2004), pp. 2347–2352.

Haddad, H., Khatib, M., Lacroix, S., and Chatila, R. 1998. Reactive navigation in outdoor environments using potential fields. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Volume 2 (1998), pp. 1232–1237 vol.2.

Hanada, M. and Ejima, Y. 2000. Effects of roll and pitch components in retinal flow on heading judgement. *Vision Research 40*, 14 (June), 1827–38.

Hartley, R. and Zisserman, A. 2000. *Multiple View Geometry in Computer Vision* (2 ed.). Cambridge University Press, Cambridge, UK.

Heeger, D. J. 1988. Optical flow using spatiotemporal filters. *International Journal of Computer Vision 1*, 279–302.

Heeger, D. J. and Jepson, A. D. 1992. Subspace methods for recovering rigid motion i: Algorithm and implementation. *International Journal of Computer Vision 7*, 2, 95–117.

Heitz, F. and Bouthemy, P. 1993. Multimodal estimation of discontinuous optical flow using markov random fields. *IEEE Transactions of Pattern Analysis and Machine Intelligence 15*, 12, 1217–1232.

Helmholtz, H. 1925. *Treatise on Physiological Optics vol. 1*. Dover: New York.

Hentout, A., Bouzouia, B., and Toukal, Z. 2007. Behaviour-based architecture for piloting mobile manipulator robots. In *Proceedings of the 2007 IEEE International Symposium on Industrial Electronics* (2007), pp. 2095–2100.

Hildreth, E. C. 1984. Computations underlying the measurement of visual motion. *Artificial Intelligence 23*, 3, 309–355.

Hong, W. and Slotine, J.-J. E. 1997. *Experimental Robotics IV*, Chapter Experiments in hand-eye coordination using active vision, pp. 130–139. Lecture Notes in Control and Information Sciences. Springer Berlin/Heidelberg.

Horn, B. K. P. and Schunck, B. G. 1981. Determining optical flow. *Artificial Intelligence 13(1-3)*, 185–203.

Horswill, I. 1993. Polly: A vision-based artificial agent. In *Proceedings of the Eleventh National Conference on Artificial Intelligence (AAAI-93)* (1993), pp. 824–829.

Horswill, M., Marrigton, S., McCullough, C., Wood, J., Pachana, N., McWilliam, J., and Raikos, M. 2008. The hazard perception ability of older drivers. *Journal of Gerontology B: Psychological sciences and social sciences 63*, 4, 212–218.

Horswill, M. and McKenna, F. 2004. Drivers' hazard perception ability: Situation awareness on the road. In S. Banbury and S. Tremblay Eds., *A Cognitive Approach to Situation Awareness*, pp. 155–75. Ashgate.

Hung, Y. S. and Ho, H. T. 1999. A Kalman filter approach to direct depth estimation incorporating surface structure. *IEEE Transactions on Pattern Analysis and Machine Intelligence 21*, 6, 570–575.

Hutchinson, S., Hager, G. D., and Corke, P. I. 1996. A tutorial on visual servo control. *IEEE Transactions on Robotics and Automation 12*, 5, 651–69.

Iida, F. 2003. Biologically inspired visual odometer for navigation of a flying robot. *Robotics and Autnomous Systems 44*, 3-4, 201–8.

IIda, F. and Lambrinos, D. 2000. Navigation in an autonomous flying robot using a biologically inspired visual odometer. In *Proceedings of Spie - the International Society for Optical Engineering*, Volume 4196 (2000), pp. 86–97.

Jain, R. 1983. Direct computation of the focus of expansion. *IEEE Transactions on Pattern Analysis and Machine Intelligence 5*, 1 (Jan.), 58–64.

Jamal, A. and Venkatesh, K. 2007. A new color based optical flow algorithm for environment mapping using a mobile robot. In *Proceedings of the IEEE International Symposium on Intelligent Control* (2007), pp. 567–572.

Jin, H., Favaro, P., and Saotto, S. 2000. Real-time 3d motion and structure of point features: a front-end system for vision-based control and interaction. In

Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Volume 2 (2000), pp. 778–779.

Joarder, K. and Raviv, D. 1994. A new method to calculate looming for autonomous obstacle avoidance. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (1994), pp. 777–780.

Kanatani, K. 1987. Structure and motion from optical flow under perspective projection. *Computer Vision Graphics and Image Processing 28*, 2, 122–146.

Kannala, J. and Brandt, S. S. 2006. A generic camera model and calibration method for conventional, wide-angle and fish-eye lenses. *IEEE Transactions on Pattern Analysis and Machine Intelligence 28*, 8, 1335–40.

Kaplan, W. 1991. *Advanced Calculus, 4th ed.* Reading, MA: Addison-Wesley.

Kelber, A. and Zeil, J. 1997. Tetragonsics guard bees interpret expanding and contracting patterns as unintended displacements in space. *Journal of Computational Physiology A 181*, 257–65.

Khatib, O. 1986. Real-time obstacle avoidance for manipulators and mobile robots. *The International Journal of Robotics Research 5*, 1, 90–98.

Koenderink, J. and Doorn, A. V. 1975. Invariant properties of the motion parallax field due to the movement of rigid bodies relative to an observer. *Optica Acta 22*, 9, 773–791.

Koenderink, J. J. and van Doorn, A. J. 1976. Local structure of movement parallax of the plane. *Journal of the Optics Society of America 66*, 7, 717–723.

Koenderink, J. J. and van Doorn, A. J. 1981. Exterospecific component of the motion parallax field. *Journal of the Optics Society of America 71*, 8, 953–957.

Kories, R. and Zimmerman, G. 1986. A versatile method for the estimation of displacement vector fields from image sequences. In *Proceedings of IEEE Workshop on Motion: Representation and Analysis* (1986), pp. 101–6.

Kundur, S. R. and Raviv, D. 1999. Novel active vision-based visual threat cue for autonomous navigation tasks. *Computer Vision and Image Understanding: CVIU 73*, 2, 169 – 182.

Kunz, C., Murphy, C., Camilli, R., Singh, H., Bailey, J., Eustice, R., Jakuba, M., Nakamura, K., Roman, C., Sato, T., Sohn, R. A., and Willis, C. 2008. Deep sea underwater robotic exploration in the ice-covered arctic ocean with auvs. In *Proceedings of the IEEE/RSJ International Conference on Robots and Intelligent Systems (IROS)* (2008), pp. 3654–3660.

Lappe, M. 2004. Building blocks for time-to-contact estimation by the brain. In H. Hecht and G. Savelsbergh Eds., *Time-to-contact*, Advances in Psychology. Amsterdam, The Netherlands: Elsevier.

Lee, D.-J., Merrell, P., and Wei, Z. 2008. Two-frame structure from motion using optical flow probability distributions for unmanned air vehicle obstacle avoidance. *Machine Vision and Applications*.

Lee, D. N. 1976. A theory of visual control of braking based on information about time to collision. *Perception 5*, 4, 437–59.

Lee, D. N. 1980. The optic flow field: the foundation of vision. *Philosophical Transactions of the Royal Society of London B 290*, 169–179.

Lee, D. N., Davies, M. N. O., Green, P. R., and van der Weel, F. R. 1993. Visual control of velocity of approach by pigeon when landing. *Journal of Experimental Biology 180*, 85–104.

Lee, D. N. and Reddish, P. E. 1981. Plummeting gannets: a paradigm of ecological optics. *Nature 293*, 293–294.

Lenser, S., Bruce, J., and Veloso, M. 2002. A modular hierarchical behavior-based architecture. In *RoboCup 2001: Robot Soccer World Cup V*, Volume 2377 of *Lecture Notes in Computer Science* (2002), pp. 79–99.

Leonard, J. J., Rikoski, R. R., Newman, P. M., and Bosse, M. 2002. Mapping partially observable features from multiple uncertain vantage points. *International Journal of Robotics Research 21*, 10-11, 943–976.

Li, H. 1992. Global interpretation of optical flow field: A least-spuares approach. In *Proceedings of the International Conference on Pattern Recognition (ICPR)* (1992), pp. 668–71.

Li, H. 2007. Two-view motion segmentation from linear programming relaxation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2007), pp. 1–8.

Lim, J. and Barnes, N. 2007. Estimation of the epipole using optical flow at antipodal points. In *Proceedings of the IEEE Workshop on Ominidirectional Vision (OMNIVIS'07)* (2007).

Lim, J. and Barnes, N. 2008. Directions of egomotion from antipodal points. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2008).

Lin, W.-Y., Tan, G.-C., and Cheong, L.-F. 2009. When discrete meets differential: Assessing the stability of structure from small motion. *International Journal of Computer Vision*.

Little, J. J., Butlhoff, H. H., and Poggio, T. 1988. Parallel optical flow using local voting. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (1988), pp. 454–9.

Liu, H., Hong, T.-H., Herman, M., Camus, T., and Chellappa, R. 1998. Accuracy vs. efficiency trade-offs in optical flow algorithms. *Computer Vision and Image Understanding: CVIU 72*, 3, 271–86.

Longuet Higgins, H. 1984. The visual ambiguity of a moving plane. *Proceedings of the Royal Society of London B223*, 165–175.

Longuet-Higgins, H. C. 1981. A computer algorithm for reconstructing a scene from two projections. *Nature 293*, 133–135.

Longuet-Higgins, H. C. and Prazdny, K. 1980. The interpretation of a moving retinal image. *Proceedings of the Royal Society of London B208*, 385–397.

Lorigo, L., Brooks, R., and Grimsou, W. 1997. Visually-guided obstacle avoidance in unstructured environments. In *Proceedings of the IEEE/RSJ International Conference on Robots and Intelligent Systems (IROS)*, Volume 1 (Sep 1997), pp. 373–379 vol.1.

Lourakis, M. I. A. and Orphanoudakis, S. C. 1999. Using planar parallax to estimate the time-to-contact. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Volume 2 (1999), pp. 640–645.

Lowe, D. G. 1999. Object recognition from local scale-invariant features. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Volume 2 (1999), pp. 1150–1157.

Lucas, B. and Kanade, T. 1981. An iterative image registration technique with an application to stereo vision. In *Proceedings of DARPA Image Understanding Workshop* (1981), pp. 121–130.

Ma, Y., Soatto, S., Kosecka, J., and Sastry, S. S. 2006. *An Invitation to 3-D Vision: From Images to geometric models* (3 ed.). Springer, New York, NY, USA.

Maddern, W. and Wyeth, G. 2008. Development of a hemispherical compound eye for egomotion estimation. In *Proceedings of the Australiasian Conference on Robotics and Automation (ACRA)* (2008).

Mandel, K. and Duffie, N. A. 1987. On-line compensation for mobile robot docking errors. *IEEE Transactions on Robotics and Automation 3*, 6, 591–8.

Marattoli, R., de Leon, C. M., Glass, T., Williams, C., Cooney, L., Berkman, I., and Tinetti, M. 1997. Driving ceasation and increased depressive symptoms: Prospective evidence from the new haven epese. Established populations for epidemiologic studies of the elderly. *Journal of the American Geriatric Society 45*, 202–6.

Marks, T., Howard, A., Bajracharya, M., Cottrell, G., and Matthies, L. 2008. Gamma-SLAM: Using stereo vision and variance grid maps for SLAM in unstructured environments. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)* (May 2008), pp. 3717–3724.

Marr, D. 1982. *Vision.* W H Freeman and Co, USA.

Matthies, L., Kanade, T., and Szeliski, R. 1989. Kalman filter-based algorithms for estimating depth from image sequences. *International Journal of Com-*

puter Vision 3, 3, 209–238.

Matthies, L., Maimone, M., Johnson, A., Cheng, Y., Willson, R., Villalpando, C., Goldberg, S., Huertas, A., Stein, A., and Angelova, A. 2007. Computer vision on mars. *International Journal of Computer Vision 75*, 1, 67–92.

Maybank, S. 1987. Apparent area of a rigid moving body. *Image and Vision Computing 5*, 111–113.

Di Marco, M., Garulli, A., Prattichizzo, D., and Vicino, A. 2003. A set theoretic approach for time-to-contact estimation in dynamic vision. *Automatica 39*, 1037–1044.

McCane, B., Novins, K., Crannitch, D., and Galvin, B. 2001. On benchmarking optical flow. *Computer Vision and Image Understanding: CVIU 84*, 1, 126–43.

McCarthy, C. and Barnes, N. 2004. Performance of optical flow techniques for indoor navigation with a mobile robot. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)* (2004), pp. 5093–8.

McCarthy, C., Barnes, N., and Mahony, R. 2008. A robust strategy for a mobile robot using flow field divergence. *IEEE Transactions on Robotics 24*, 4, 832–842.

McQuirk, I. S., Horn, B. K. P., Lee, H.-S., and Wyatt, J. L. 1998. Estimating the focus of expansion in analog VLSI. *International Journal of Computer Vision 28*, 3, 261–277.

Meyer, F. G. 1994. Time-to-collision from first-order models of the motion field. *IEEE Transactions on Robotics and Automation 10*, 6, 792–8.

Micusik, B., Wildenauer, H., and Vincze, M. 2008. Towards detection of orthogonal planes in monocular images of indoor environments. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)* (2008), pp. 999–1004.

Milford, M. and Wyeth, G. 2008. Single camera vision-only SLAM on a suburban road network. In *Proceedings of the IEEE International Conference on Robotics* *and Automation (ICRA)* (May 2008), pp. 3684–3689.

Mitiche, A. and Sekkati, H. 2006. Optical flow 3D segmentation and interpretation: A variational method with active curve evolution and level sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence 28*, 11, 1818–1829.

Mochizuki, Y., Imiya, A., and Torii, A. 2007. Circle-marker detection method for omnidirectional images and its application to robot positioning. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (2007), pp. 1–8.

Montemerlo, M., Thrun, S., Koller, D., and Wegbreit, B. 2003. Fast-SLAM 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges. In *Proceedings of the International Joint Conference on Artificial Intelligence* (2003), pp. 1151–1156.

Moravec, H. P. 1983. The stanford cart and the cmu rover. *Proceedings of the IEEE 71*, 7, 872–884.

Moravec, H. P. and Elfes, A. 1985. High resolution maps from wide angle sonar. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)* (1985), pp. 116–121.

Mouragnon, E., Lhuillier, M., Dhome, M., Dekeyser, F., and Sayd, P. 2006. 3D reconstruction of complex structures with bundle adjustment: an incremental approach. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)* (2006), pp. 3055–3061.

Murray, D. and Little, J. J. 2000. Using real-time stereo vision for mobile robot navigation. *Autonomous Robots 8*, 2, 161–171.

Nagel, H.-H. 1990. Extending the ‘oriented smoothness constraint’ into the temporal domain and the estimation of derivatives of optical flow. In *Proceedings of the 1990 European Conference on Computer Vision (ECCV). Lecture Notes in Computer Science, Computer Vision*, Volume 427 (1990).

Negahdaripour., S. 1996. Direct computation of the FOE with confidence measures. *Computer Vision and Image Understanding: CVIU 64*, 3, 323–350.

Negahdaripour, S. and Horn, B. K. P. 1989. A direct method for locating the focus of expansion. *Computer Vision Graphics and Image Processing 46*, 3, 303–326.

Nelson, R. C. and Alloimonos, J. Y. 1989. Obstacle avoidance using flow field divergence. *IEEE Transactions on Pattern Analysis and Machine Intelligence 11*, 10, 1102–6.

Nelson, R. C. and Aloimonos, J. 1988. Finding motion parameters from spherical motion fields (or the advantages of having eyes in the back of your head). *Biological Cybernetics 58*, 261–73.

Neppi-Mòdona, M., Auclair, D., Sirigu, A., and Duhamel, J.-R. 2004. Spatial coding of the predicted impact location of a looming object. *Current Biology 14*, 1174–1180.

Ng, L. and Solo, V. 2001. Errors-in-variables modeling in optical flow estimation. *IEEE Transactions on Image Processing 10*, 10, 1528–1540.

Nierobisch, T., Fischer, W., and Hoffmann, F. 2006. Large view visual servoing of a mobile robot with a pan-tilt camera. In *Proceedings of the IEEE/RSJ International Conference on Robots and Intelligent Systems (IROS)* (2006), pp. 3307–3312.

Nilsson, N. 1969. A mobile automaton: An application of artificial intelligence techniques. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI-69)* (1969), pp. 509–20.

Nilsson, N. 1984. Shakey the robot. Technical Report No. 323, Artificial Intelligence Center, SRI International, Menlo Park, CA.

Nister, D. 2003. Preemptive RANSAC for live structure and motion estimation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Volume 1 (2003), pp. 199–206.

Nister, D. 2005. Preemptive RANSAC for live structure and motion estimation. *Machine Vision and Applications 16*, 5, 321–329.

Nister, D., Naroditsky, O., and Bergen, J. 2004. Visual odometry. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*

(CVPR), Volume 1 (2004), pp. 652–659.

Ogino, M., Kikuchi1, M., and Asada, M. 2006. Visuo-motor learning for face-to-face pass between heterogeneous humanoids. *Robotics and Autonomous Systems 54*, 6, 419–427.

Oliensis, J. 2005. The least-squares error for structure from infinitesimal motion. *International Journal of Computer Vision 61*, 3, 259–299.

Olson, C. F., Matthies, L. H., Wright, J. R., Li, R., and Di, K. 2007. Visual terrain mapping for mars exploration. *Computer Vision and Image Understanding 105*, 1, 73–85.

Otte, M. W., Richardson, S. G., Mulligan, J., and Grudic, G. 2007. Local path planning in image space for autonomous robot navigation in unstructured environments. In *Proceedings of the IEEE/RSJ International Conference on Robots and Intelligent Systems (IROS)* (2007), pp. 2819–2826.

Overett, G., Petersson, L., Andersson, L., and Pettersson, N. 2009. Boosting a heterogeneous pool of fast HOG features for pedestrian and sign detection. In *Proceedings of the IEEE Intelligent Vehicles Symposium* (2009).

Pacheco, L., Cufi, X., Luo, N., and Cobos, J. 2008. WMR navigation using local potential field corridors and narrow local occupancy grid perception. In *IEEE International Conference on Automation, Quality and Testing, Robotics*, Volume 2 (2008), pp. 304–309.

Pollefeys, M., Gool, L. V., Vergauwen, M., Verbiest, F., Cornelis, K., Tops, J., and Koch, R. 2004. Visual modeling with a hand-held camera. *International Journal of Computer Vision 59*, 207–232.

Pollefeys, M., Nister, D., Frahm, J.-M., Akbarzadeh, A., Mordohai, P., Clipp, B., Engels, C., Gallup, D., Kim, S.-J., Merrell, P., Salmi, C., Sinha, S., Talton, B., Wang, L., Yang, Q., Stewenius, H., Yang, R., Welch, G., and Towles, H. 2008. Detailed real-time urban 3d reconstruction from video. *International Journal of Computer Vision 78*, 2 (July), 143–167.

Questa, P., Grossmann, E., and Sandini, G. 1995. Camera self orientation and docking maneuver using normal flow. In *Proceedings of Spie - the International Society for Optical Engineering*, Volume 2488 (1995), pp. 274–83.

Questa, P. and Sandini, G. 1996. Time to contact computation with a space-variant retina-like c-mos sensor. In *Proceedings of the IEEE/RSJ International Conference on Robots and Intelligent Systems (IROS)*, Volume 3 (1996), pp. 1622–1629.

Regan, D. and Beverly, K. I. 1982. How do we avoid confounding the direction we are looking and the direction we are moving. *Science 215*, 194–196.

Reichardt, W. 1969. Movement perception in insects. In W. Reichardt Ed., *Processing of Optical Data by Organizms and by Machines*, pp. 465–493. New York: Academic.

Rieger, J. H. and Lawton, H. T. 1985. Processing differential image motion. *Journal of the Optics Society of America - A 2*, 2, 354–60.

Rind, F. C. 1997. Collision avoidance: from the locust eye to a seeing machine. In M. V. Srinivasan and S. Venkatesh Eds., *From Living Eyes to Seeing Machines*, pp. 105–125. Oxford UK: Oxford University Press.

Roberts, J., Duff, E., and Corke, P. 2003. Automation of an underground mining vehicle using reactive navigation and opportunistic localization. In *Proceedings of the IEEE/RSJ International Conference on Robots and Intelligent Systems (IROS)*, Volume 3 (2003), pp. 3775–3780.

Roberts, J. M., Corke, P. I., and Buskey, G. 2003. Low-cost flight control system for a small autonomous helicopter. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)* (2003), pp. 546–52.

Robertson, R. M. and Johnson, A. G. 1993. Collision avoidance of flying locusts: steering torques and behaviour. *Journal of Experimental Biology 183*, 35–60.

Ruffier, F. and Franceschini, N. 2005. Optic flow regulation: the key to aircraft automatic guidance. *Robotics and Autnomous Systems 50*, 177–194.

Santos-Victor, J. and Sandini, G. 1994. Visual behaviors for docking. Technical Report LIRA-TR 2/94 (June), LIRA Lab, University of Genoa.

Santos-Victor, J. and Sandini, G. 1995. Divergent stereo in autonomous navigation: From bees to robots. *International Journal of Computer Vision 14*, 2, 159–77.

Santos-Victor, J. and Sandini, G. 1996. Uncalibrated obstacle detection using normal flow. *Machine Vision and Applications 9*, 3, 130–37.

Santos-Victor, J. and Sandini, G. 1997. Visual behaviors for docking. *Computer Vision and Image Understanding: CVIU 67*, 3, 223–38.

Sazbon, D., Rotstein, H., and Rivlin, E. 2004. Finding the focus of expansion and estimating range using optical flow images and a matched filter. *Machine Vision and Applications 15*, 229–36.

Schiff, W., Caviness, J. A., and Gibson, J. J. 1962. Persistent fear responses in rhesus monkeys to the optical stimulus of looming. *Science 136*, 982–3.

Segvic, S., Remazeilles, A., Diosi, A., and Chaumette, F. 2007. Large scale vision-based navigation without an accurate global reconstruction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2007), pp. 1–8.

Shao, Y., Mayhew, J., and Hippisley Cox, S. 1995. Ground plane obstacle detection of stereo vision under variable camera geometry using neural nets. In *Proceedings of the British Machine Vision Conference (BMVC)* (1995).

Sharp, C. S., Shakernia, O., and Sastry, S. S. 2001. A vision system for landing an unmanned aerial vehicle. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)* (2001), pp. 1720–7.

Simoncelli, E. P., Adelson, E. H., and Heeger, D. J. 1991. Probability distributions of optical flow. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (1991), pp. 310–315.

Smith, R. and Cheesman, P. 1987. On the representation of spatial uncertainty. *International Journal of Robotics Research 5*, 4, 56–68.

Song, K.-T. and Chen, H.-Y. 2007. Lateral driving assistance using optical flow and scene analysis. In *Proceedings of the 2007 IEEE Intelligent Vehicles Symposium* (2007), pp. 624–9.

Srinivasan, M. V. and Zhang, S. W. 2004. Visual motor computations in insects. *Annual Review of Neuroscience 27*, 679–696.

Srinivasan, M. V., Zhang, S. W., Chahl, J. S., Barth, E., and Venkatesh, S. 2000. How honeybees make grazing landings on flat surfaces. *Biological Cybernetics 83*, 171–83.

Srinivasan, S. 1999. Fast partial search solution to the 3d sfm problem. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Volume 1 (1999), pp. 528–535.

Srinivasan, S. 2000. Extracting structure from optical flow using the fast error search technique. *International Journal of COmputer Vision 37*, 3 (June), 203–230.

Subbarao, M. 1989. Interpretation of image flow: a spatio-temporal approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence 2*, 3, 266–278.

Subbarao, M. 1990. Bounds on time-to-collision and rotational component from first-order derivatives of image flow. In *Computer Vision, Graphics and Image Processing*, Volume 50 (1990), pp. 329 – 41.

Subbarao, M. and Waxman, A. 1986. Closed form solutions to image flow equations for planar surfaces in motion. *Computer Vision Graphics and Image Processing 36*, 208–228.

Sutton, M. A., Walters, W. J., Peters, W. H., Ranson, W. F., and McNeil, S. R. 1983. Determination of displacement using an improved digital correlation method. *Image and Vision Computing 1*, 3, 133–9.

Suzuki, T. and Kanade, T. 1999. Measurement of vehicle motion and orientation using optical flow. In *Proceedings of the 1999 IEEE International Conference on Intelligent Transport Systems* (1999), pp. 25–30.

Tho, H. H. and Goecke, R. 2008. Optical flow estimation using fourier mellin transform. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2008), pp. 1–8.

Thorpe, C., Hebert, M., Kanade, T., and Shafer, S. 1988. Vision and navigation for the carnegie-mellon navlab. *IEEE Transactions on Pattern Analysis and*

Machine Intelligence 10, 3, 362–373.

Thrun, S. 1998. Bayesian landmark learning for mobile robot localization. *Machine Learning 33*, 1, 41–76.

Thrun, S., Burgard, W., and Fox, D. 2000. A real-time algorithm for mobile robot mapping with applications to multi-robot and 3d mapping. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Volume 1 (2000), pp. 321–328 vol.1.

Thrun, S., Burgard, W., and Fox, D. 2005. *Probabalistic Robotics*. The MIT Press, Cambridge, Massachusetts, USA.

Tistarelli, M. and Sandini, G. 1993. On the advantages of polar and log-polar mapping for direct estimation of time-to-impact from optical flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence 15*, 4, 401–10.

Triggs, B., McLauchlan, P., Hartley, R., and Fitzgibbon, A. 2000. Bundle adjustment-a modern synthesis. In *Vision Algorithms: Theory and Practice: Springer Lecture Notes on Computer Science*, Volume 1883 (Berlin Heidelberg New York, 2000), pp. 298–375. Springer Verlag.

Tsai, R. Y. and Huang, T. S. 1981. Estimating three-dimensional motion parameters of a rigid planar patch. *IEEE Transactions on Acoustics, Speech and Signal Processing 29*, 6, 1147–1152.

Ulrich, I. and Borenstein, J. 1998. VFH+: reliable obstacle avoidance for fast mobile robots. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Volume 2 (1998), pp. 1572–1577.

Ulrich, I. and Borenstein, J. 2000. VFH: local obstacle avoidance with look-aheadverification. In *Proceedings of the 2000 IEEE International Conference on Robotics and Automation*, Volume 3 (2000), pp. 2505–2511.

Urmson, C. P., Dias, M. B., and Simmons, R. G. 2002. Stereo vision based navigation for sun-synchronous exploration. In *Proceedings of the IEEE/RSJ International Conference on Robots and Intelligent Systems (IROS)*, Volume 1 (2002), pp. 805–810 vol.1.

Usher, K., Ridley, P., and Corke, P. 2003. Visual servoing of a car-like vehicle - an application of omnidirectional vision. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)* (Taipei, Taiwan, 2003), pp. 4288–93.

van Leeuwen, M. B. and Groen, F. C. A. 2000. Motion estimation wth a mobile camera for traffic applications. In *Proceedings of the IEEE Intelligent Vehicles Symposium 2000* (2000), pp. 58–63.

van Leeuwen, M. B. and Groen, F. C. A. 2002. Motion interpretation for in-car vision systems. In *Proceedings of the IEEE/RSJ International Conference on Robots and Intelligent Systems (IROS)* (2002), pp. 135–40.

Verri, A. and Poggio, T. 1989. Motion field and optical flow: qualitative properties. *IEEE Transactions on Pattern Analysis and Machine Intelligence 11*, 5, 490–498.

Vidal, R. and Hartley, R. 2004. Motion segmentation with missing data using PowerFactorization and GPCA. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (June 2004), pp. II–310 – 316.

Vidal, R. and Hartley, R. 2008. Three-view multibody structure from motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence 30*, 2 (Feb.), 214–227.

Wagner, H. 1982. Flow-field variables trigger landing in flies. *Nature 297*, 147–148.

Warren, C. W. 1989. Global path planning using artificial potential fields. *IEEE Transactions on Robotics and Automation 1*, 316–321.

Watson, A. B. and Ahumada, A. J. 1983. A look at motion in the frequency domain. In J. K. Tsotsos Ed., *Motion: Perception and representation*, pp. 1–10.

Waxman, A. M. and Ullman, S. 1985. Surface structure and three-dimensional motion from image flow kinematics. *The International Journal of Robotics Research 4*, 3, 72–94.

Weber, J. and Malik, J. 1993. Robust computation of optical flow in a multi-scale differential framework. In *Proceedings of the IEEE International Conference on*

Computer Vision (ICCV) (1993), pp. 12–20.

Weber, J. and Malik, J. 1997. Rigid body segmentation and shape description from dense optical flow under weak perspective. *IEEE Transactions on Pattern Analysis and Machine Intelligence 19*, 2, 139–43.

Weber, K., Venkatesh, S., and Srinivasan, M. V. 1996. Insect inspired behaviours for the autonomous control of mobile robots. In *Proceedings of the International Conference on Pattern Recognition (ICPR)* (Vienna, Austria, 1996), pp. 156–60.

Wedel, A., Franke, U., Badino, H., and Cremers, D. 2008. B-spline modeling of road surfaces for freespace estimation. In *Proceedings of the 2008 IEEE Intelligent Vehicles Symposium* (2008), pp. 828–833.

Wei, R., Austin, D., and Mahony, R. 2005. Biomimetic application of desert ant visual navigation for mobile robot docking with weighted landmarks. *International Journal of Intelligent Systems Technologies and Applications 1*, 1/2, 174–190.

Weickert, J. and Schnörr, C. 2001. Variational optic flow computation with a spatio-temporal smoothness constraint. *Journal of Mathematical Imaging and Vision 14*, 3, 245–255.

Williams, S. and Howard, A. 2008. A single camera terrain slope estimation technique for natural arctic environments. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)* (2008), pp. 2729–2734.

Yamaguchi, K., Kato, T., and Ninomiya, Y. 2006. Vehicle ego-motion estimation and moving object detection using a monocular camera. In *Proceedings of the International Conference on Pattern Recognition (ICPR)*, Volume 4 (0-0 2006), pp. 610–613.

Young, G.-S. and Chellappa, R. 1992. Statistical analysis of inherent ambiguities in recovering 3-d motion from a noisy flow field. *IEEE Transactions on Pattern Analysis and Machine Intelligence 14*, 10 (Oct), 995–1013.

Zako, M., McIntyre, J., Senot, P., and Lacquaniti, F. 2009. Visuo-motor coordination and internal models for object interception. *Experimental Brain Re-*

search 192, 571–604.

Zhang, H. and Ostrowski, J. P. 2002. Visual motion planning for mobile robots. *IEEE Transactions on Robotics and Automation 18*, 2, 199–208.

Zhou, J. and Li, B. 2006. Homography-based ground detection for a mobile robot platform using a single camera. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)* (2006), pp. 4100–4105.

Zufferey, J.-C. and Floreano, D. 2006. Fly-inspired visual steering of an ultralight indoor aircraft. *IEEE Transactions on Robotics 22*, 1, 137–146.

Zwaan, S., Bernardino, A., and Santos-Victor, J. 2002. Visual station keeping for floating robots in unstructured environments. *Robotics and Autnomous Systems 39*, 3–4, 145–55.