Linköping studies in science and technology. Dissertations.
No. 1623

Inertial Navigation and Mapping for Autonomous Vehicles

Martin Skoglund

Linköping University

Department of Electrical Engineering
Linköping University, SE-581 83 Linköping, Sweden

Linköping 2014

Cover illustration: The cover illustration is left intentionally black!

Linköping studies in science and technology. Dissertations.
No. 1623

Inertial Navigation and Mapping for Autonomous Vehicles

Martin Skoglund

ms@isy.liu.se
www.control.isy.liu.se
Division of Automatic Control
Department of Electrical Engineering
Linköping University
SE-581 83 Linköping
Sweden

ISBN 978-91-7519-233-8

ISSN 0345-7524

Copyright © 2014 Martin Skoglund

Printed by LiU-Tryck, Linköping, Sweden 2014

To Maria

Abstract

Navigation and mapping in unknown environments is an important building block for increased autonomy of unmanned vehicles, since external positioning systems can be susceptible to interference or simply being inaccessible. Navigation and mapping require signal processing of vehicle sensor data to estimate motion relative to the surrounding environment and to simultaneously estimate various properties of the surrounding environment. Physical models of sensors, vehicle motion and external influences are used in conjunction with statistically motivated methods to solve these problems. This thesis mainly addresses three navigation and mapping problems which are described below.

We study how a vessel with known magnetic signature and a sensor network with magnetometers can be used to determine the sensor positions and simultaneously determine the vessel's route in an *extended Kalman filter* (EKF). This is a so-called *simultaneous localisation and mapping* (SLAM) problem with a reversed measurement relationship.

Previously determined hydrodynamic models for a *remotely operated vehicle* (ROV) are used together with the vessel's sensors to improve the navigation performance using an EKF. Data from sea trials is used to evaluate the system and the results show that especially the linear velocity relative to the water can be accurately determined.

The third problem addressed is SLAM with inertial sensors, accelerometers and gyroscopes, and an optical camera contained in a single sensor unit. This problem spans over three publications.

We study how a SLAM estimate, consisting of a point cloud map, the sensor unit's three dimensional trajectory and speed as well as its orientation, can be improved by solving a *nonlinear least-squares* (NLS) problem. NLS minimisation of the predicted motion error and the predicted point cloud coordinates given all camera measurements is initialised using EKF-SLAM.

We show how NLS-SLAM can be initialised as a sequence of almost uncoupled problems with simple and often linear solutions. It also scales much better to larger data sets than EKF-SLAM. The results obtained using NLS-SLAM are significantly better using the proposed initialisation method than if started from arbitrary points. A SLAM formulation using the *expectation maximisation* (EM) algorithm is proposed. EM splits the original problem into two simpler problems and solves them iteratively. Here the platform motion is one problem and the landmark map is the other. The first problem is solved using an extended Rauch-Tung-Striebel smoother while the second problem is solved with a quasi-Newton method. The results using EM-SLAM are better than NLS-SLAM both in terms of accuracy and complexity.

Populärvetenskaplig sammanfattning

Vi människor utför dagligen en stor mängd navigerings- och karteringsrutiner utan att ens reflektera över dessa. Allt ifrån att resa mellan hem och arbete till att upptäcka nya spännande miljöer i en ny stad. För navigering i välbekanta miljöer använder vi våra sinnen och den mentala karta av omvärlden som är uppbyggd av tidigare upplevelser. Att navigera i okända miljöer är svårare eftersom vi måste strukturera en ny mental karta av alla intryck och samtidigt uppfatta var i kartan vi befinner oss. Lyckligtvis finns en stor mängd hjälpmedel såsom, kartor, kompasser, satellitpositioneringssystem, med mera, som underlättar för oss.

Navigering och kartering i okända miljöer är även en viktig byggsten för obemannade farkosters ökade autonomi eftersom externa positioneringssystem kan vara störningskänsliga eller helt enkelt otillgängliga. Navigering och kartering kräver, ofta omfattande, signalbehandling av farkosternas sensordata för att uppskatta hur den rör sig relativt omvärlden och att samtidigt uppskatta olika egenskaper i omvärlden. Fysikaliska modeller av sensorer, farkosters rörelse och yttre påverkan används tillsammans med statistiskt motiverade metoder för att lösa dessa problem så bra som möjligt. Denna avhandling behandlar huvudsakligen tre navigerings- och karteringsproblem som beskrivs nedan.

Säkerhetskritiska maritima miljöer, såsom hamnar, kylvattenanläggningar vid kärnkraftverk och andra skyddsobjekt kräver ständig övervakning. För att detektera och göra målföljning av främmande fartyg och undervattensfarkoster i dessa miljöer kan man använda sig av sensorer placerade i ett nätverk på havsbotten. Dessa nätverk fungerar endast tillförlitligt om sensorernas position och orientering på havsbotten är känd. Vi studerar hur ett fartyg med känd magnetisk signatur och ett sensornätverk med tre-axliga magnetometrar kan användas för att bestämma sensorernas position och samtidigt bestämma fartygets rutt med ett så kallat *extended Kalman filter* (EKF). Detta är ett så kallat samtidig lokalisering och karteringsproblem, på engelska *simultaneous localisation and mapping* (SLAM), men med omvänd mätrelationen. Analys av hur noggrant sensorernas position kan bestämmas för en given rutt redovisas och det omvända, alltså hur noggrant en rutt kan bestämmas när sensorernas positioner är kända. Dessutom visas med känslighetsanalys att om fartyget är utrustat med positioneringssystem så kan fel i sensororientering och fel i den magnetiska signaturen undertryckas och positionsnoggrannheten för sensorerna förbättras. Systemet utvärderas med hjälp av simuleringar.

Det andra problemet som behandlas är modellering en fjärrstyrd obemannad undervattensfarkost, på engelska *remotely operated vehicle* (ROV), och sensorfusion. ROVar används i en mängd krävande undervattenstillämpningar såsom tunnel- och skrovinspektion, svetsning på oljeriggar där djupet är för stort för vanliga dykare och arkeologiska expeditioner. ROVar har ett begränsat utrymme för navigerings- och karteringssensorer och dessa måste även vara relativt billiga. Dessutom saknas det ofta externa positioneringssystem under vattnet och därför krävs oberoende robusta navigeringsmetoder. Vi studerar hur tidigare bestämda

hydrodynamiska delmodeller för en ROV kan användas tillsammans med farkostens sensorer för att förbättra navigeringsprestanda. De hydrodynamiska modellerna beskriver farkostens rörelse i det omgivande vattnet där rörelsen genereras av de fem propellrarna. Vi redovisar modeller för alla sensorer uttryckta i ett kroppsfixt system och dessa används tillsammans med de hydrodynamiska modellerna i ett EKF. Experimentella data från sjökörningar i Vättern används för att utvärdera systemet och resultaten visar att framför allt linjär hastighet relativt vattnet kan bestämmas noggrant.

Det tredje problemet som behandlas är SLAM med tröghetssensorer (accelerometrar och gyroskop) och optisk kamera i en sammansatt sensormodul. Detta problem sträcker sig över tre publikationer.

I den första publikationen studerar vi hur en SLAM-skattning, bestående av en skalenligt punktmolnskarta, sensormodulens tre-dimensionella bana och hastighet samt dess orientering, kan förbättras genom att lösa ett olinjärt minstakvadratproblem. Den initiala SLAM-skattningen görs med EKF-SLAM och det förbättrade estimated fås genom att minimera det predikterade rörelsefelet över hela banan och de predikterade punktmolnskoordinaterna givet alla kameramätningar och där tröghetssensorerna behandlas som en känd insignal. Resultaten är utvärderade med väldigt tillförlitliga data där sensormodulen är fastspänd i verktygspositionen på en industrirobot och miljön uppmätt för hand.

I den andra publikationen visar vi hur det olinjära minstakvadratproblemet för SLAM kan initialiseras med en sekvens av nästan frikopplade problem med enkla, och oftast linjära, lösningar. Fördelen med att använda denna metod, istället för EKF-SLAM, är att den kan användas på mycket större dataset. Vi visar även att resultatet från det olinjära minstakvadratproblemet blir bättre med den föreslagna initialiseringsmetoden än om den startas från godtyckliga punkter. Flera steg i metoden utvärderas med Monte Carlo simuleringar och experimentella data.

I den tredje publikationen studeras en alternativ formulering till det olinjära minstakvadratproblemet med hjälp av *expectation maximisation* (EM) algoritmen. EM är en metod som används för att hantera komplexa problem genom att dela upp det i två enklare problem och lösa dessa iterativt. I den föreslagna metoden betraktas sensormodulens bana som det ena problemet och det andra problemet utgörs av punktmolnskartan. EM metoden initialiseras med den ovan beskrivna metoden och resultat på experimentell och simulerad data visar sig ge mindre medelfel än med den olinjära minstakvadratmetoden. Dessutom är metoden beräkningsmässigt mer effektiv.

Acknowledgments

The quote “*I travel light. But not at the same speed.*” – Jarod Kintz, perfectly summarises that my long journey has *finally* reached its end. This would of course not have been possible without the support and encouragement from colleagues, friends and family. My first acknowledgement is directed to my supervisor Prof Fredrik Gustafsson, your inspiring tempo, professional sharpness and relaxed attitude to the never-ceasing flood of tight deadlines is impressive. Thanks for dragging me along this fascinating ride. Thanks! I also like to thank my co-supervisor Prof Thomas B Schön. Thomas, you have a very professional attitude and is an excellent researcher and reviewer with a bunch of cool estimation skills up your sleeve. Thanks! Various parts of this thesis has also been proof-read by Dr Zoran Sjanic and Dr Gustaf Hendeby. Thanks for all the comments and letting me borrow your time! All remaining eRrors are mine! Thanks to Dr Gustav Hendeby and Dr Henrik Tidefelt for help with figures in *Shapes* and L^AT_EX questions and for swiftly improving the thesis template. Working at the Division of Automatic Control has been a great experience. Thanks to all the nice colleagues and the curious students! Prof Svante Gunnarsson responsively manages the division creating a nice atmosphere, thanks! Our secretary Ninna Stensgård, and her two predecessors Åsa Karmelind and Ulla Salaneck make sure that administrative and practical matters work flawlessly. Thank you!

Thanks to the Industry Excellence Center LINK-SIC founded by Vinnova for financial support. A special thanks Dr Martin Enqvist for managing everything related to LINK-SIC smoothly.

Dr Zoran Sjanic, it's been great working together and I hope we can continue on this quest! Besides being a clever guy, you are also a good friend with a funky sense of humor that I enjoy. Thanks!

Dr Magnus Axholt, your super positive attitude was vital when we worked with HMD calibration. I also really enjoyed learning more about augmented reality and optical systems. I am really impressed by your entrepreneurial skills! Thanks!

Dr Jonas Callmer and I have shared many hours of studies and a bizarre amount of coffee whilst quoting South Park. Unfortunately we don't meet as often since you joined *the dark side* outside university. That will hopefully change when you sell SenionLab for a trillion balubas!

Dr Karl Granström –*master of beer induced discussions* – and I shared office for the first years of the PhD studies. Thanks for all the good times we have had at RT and outside work!

Dr David Törnqvist gave me a smooth start in the group when supervising my Master's Thesis. You have been my next-door neighbour in the corridor, fellow musician, dive mate and a good friend. Thanks David!

Thanks Dr Karl-Johan Thore for being an excellent study companion and friend.

I'm happy that you finally moved to the right side of the city!

Johan Reunanen, you inspired me to move to Linköping to study and I also joined your *vaniljekrem*-gang in Oslo. We got to know each other when studying a preparatory year of Natural Sciences in our hometown Sundsvall and I hope we get to meet more often.

Johan Bjugert experimentally proved that conserving eggs in the freezer is a bad idea! It was fun living together in Ryde on BG's second floor. I hope we get to meet more often now when you are back in Sweden. Do you still prepare an extra serving of Pasta Carbonara for your trumpet?

Thanks to my old friend Jens *struuut*! Södervall for all these years!

Thanks Per *Buffy* Keller for all the fun in Dublin and elsewhere!

Clas Veibäck, it's nice to have an office mate again. Especially with someone who is as eager as I to put on a Maalouf album on our blazing sound system!

Thanks to Henrik Tidefelt and Nina Fjällström for all the fun we have together! Congratulations to the marriage and for letting us be a part of your day!

These years have been accompanied by a lot of great music. Thanks to Miles Davis, Håkan Hardenberger, Ibrahim Maalouf, Wynton Marsalis, Esbjörn Svensson Trio, Goran Kajfeš, Wolfgang Amadeus and Leopold Mozart, Igor Stravinsky, Brad Mehldau, Al Di Meola, Avishai Cohen, Pat Metheny, Keith Jarrett and the great Ludwig van.

Over the years I have made many friends. Dr André Carvalho Bittencourt, Lic Sina Khoshfetrat Pakazad, Dr Patrik Axelsson, Lic Ylva Ljung, Dr Christian Lundquist, Dr Per Skoglar, Dr Daniel Peterson, Dr Christian Lyzell, Dr Johan Sjöberg, Dr Jeroen Hol, Dr Gustaf Hendby, Lic Manon Kok, Dr Emre Özkan, Dr Saikat Saha, Dr Daniel Axehill, Jonas Linder, Dr Fredrik Lindsten, Lic Michael Roth and possibly more people. Thank you for making my life much more enjoyable!!

Outside work I have had particularly many great laughs together with Dr Hanna Fager, Dr Oskar Leufvén, Dr Jonas Callmer and Dr Karl Granström. Thanks!

Thanks to the Falkenberg people for all the fun we have together! Thanks to my late mother Helena for all love, care and the somewhat useful stubbornness I inherited. My brother John and father Bernt, I hope you feel as excited and proud as I do! Thanks for everything!

Finally, thanks and a thousand kisses to Maria, Stella and the unborn *lill-räpen* for all your love and support!

Linköping, September 2014
Martin Skoglund

Contents

Notation	xv
----------	----

I Background

1 Introduction	3
1.1 Motivation	3
1.2 Contributions	5
1.2.1 Included Contributions	7
1.2.2 Additional Publications	10
1.2.3 Optical See-Through Head Mounted Display Calibration	10
1.3 Thesis Outline	12
2 Models	15
2.1 Kinematics	16
2.1.1 Translational Kinematics	16
2.1.2 Rotational Kinematics	17
2.1.3 Rigid Body Kinematics	20
2.2 Rigid Body Dynamics	21
2.3 Inertial Sensors	22
2.3.1 Gyroscopes	22
2.3.2 Accelerometers	23
2.4 Magnetometers	23
2.5 Vision Sensors	24
2.5.1 The Pinhole Camera	24
2.6 Camera Measurement Models	27
2.6.1 Direct Parametrisation	27
2.6.2 Inverse Depth	28
2.6.3 Epipolar Geometry	29
2.7 Computer Vision	34
3 Estimation	37
3.1 Sensor Fusion	37

3.1.1	Smoothing and Filtering	38
3.2	Optimisation	39
3.2.1	Iterated Extended Kalman Filter	41
3.2.2	Nonlinear Least Squares	44
3.3	Problem Structure	46
4	SLAM	49
4.1	Introduction	49
4.1.1	Probabilistic Models	49
4.2	EKF-SLAM	51
4.3	Batch SLAM	53
4.3.1	Graphs	53
4.3.2	Initialisation	55
4.3.3	NLS-SLAM	56
4.3.4	EM-SLAM	57
4.4	IMU and Camera	58
4.4.1	Linear Triangulation	59
5	Concluding remarks	63
5.1	Conclusions	63
5.2	Future Work	64
	Bibliography	65

II Publications

A	Silent Localization of Underwater Sensors using Magnetometers	81
1	Introduction	83
2	Methodology	85
2.1	System Description	86
2.2	State Estimation	87
2.3	Cramér-Rao Lower Bound	89
3	Simulation Results	90
3.1	Magnetometers Only	90
3.2	Magnetometers and GNSS	91
3.3	Trajectory Evaluation using CRLB	92
3.4	Sensitivity Analysis, Magnetic Dipole	93
3.5	Sensitivity Analysis, Sensor Orientation	94
4	Conclusions	96
	Bibliography	99
B	A Nonlinear Least-Squares Approach to the SLAM Problem	103
1	Introduction	105
2	Problem Formulation	106
3	Models	107
3.1	Dynamics	108

3.2	Landmark State Parametrisation	108
3.3	Camera Measurements	109
4	Solution	110
4.1	Initialisation	110
4.2	Nonlinear Least-Squares Smoothing	111
5	Experiments	114
5.1	Experimental Setup	114
5.2	Results	114
6	Conclusions and Future Work	115
	Bibliography	118
C	Modeling and Sensor Fusion of a Remotely Operated Underwater Vehicle	121
1	Introduction	123
2	Related Work	124
3	System Overview	126
3.1	Kinematics	127
4	ROV Modeling	127
4.1	Hydrodynamic Models	128
4.2	Discretization	131
4.3	Kinematic model	131
5	Sensor Models	132
5.1	Inertial Measurement Unit and Magnetometer	132
5.2	Doppler Velocity Log	133
6	Estimation	134
6.1	Filtering	134
6.2	Synchronization	135
7	Results	135
7.1	Velocity Model	136
7.2	Angular Velocity Model	136
7.3	Position Estimates	137
8	Conclusion	137
	Bibliography	139
D	Initialisation and Estimation Methods for Batch Optimisation of Inertial/Visual SLAM	143
1	Introduction	145
2	Models	149
2.1	Position and Orientation	149
2.2	IMU Measurements	150
2.3	Camera Measurements	150
3	SLAM Initialisation	151
3.1	Feature Tracks	151
3.2	Track Clustering	153
3.3	Rotation Initialisation	155
3.4	Linear SLAM	155

	3.5	Iterative Outlier Removal	158
4		Nonlinear Least-Squares SLAM	159
5		Heuristic Motivation of the Linear Initialisation	159
6		Monte Carlo Simulations	160
	6.1	Efficiency of the Linear Initialisation	161
	6.2	Sensitivity to Initial Rotation Errors	161
	6.3	Iterative Outlier Removal	162
7		Real Data Experiments	164
	7.1	Clustering Results	165
8		Conclusions and Future Work	171
		Bibliography	176
E		EM-SLAM with Inertial/Visual Applications	181
1		Introduction	183
2		Expectation Maximisation	185
3		EM-SLAM	186
	3.1	E-step	187
	3.2	M-step	188
4		Models	188
	4.1	IMU Parametrisation	190
	4.2	Camera Measurements	190
5		Nonlinear Least-Squares	191
6		Computation Complexity	192
7		Obtaining an Initial Estimate	193
8		Results	194
	8.1	Simulations	194
	8.2	Real Data Experiments	196
9		Conclusions and Future Work	196
		Bibliography	201

Notation

ABBREVIATIONS

Abbreviation	Meaning
BA	<i>bundle adjustment</i>
CRLB	<i>Cramér-Rao lower bound</i>
CCD	<i>charge coupled devices</i>
CMOS	<i>complementary metal oxide semiconductor</i>
DOF	<i>degrees of freedom</i>
DVL	<i>Doppler velocity log</i>
EKF	<i>extended Kalman filter</i>
EM	<i>expectation-maximisation</i>
E-RTS	<i>extended Rauch-Tung-Striebel smoother</i>
GNSS	<i>global navigations satellite system</i>
GPS	<i>global positioning system</i>
IMU	<i>inertial measurement unit</i>
IDP	<i>inverse depth parametrisation</i>
KF	<i>Kalman filter</i>
LiDAR	<i>light detection and ranging</i>
MAP	<i>maximum a posteriori</i>
MEMS	<i>micro electrical mechanical systems</i>
ML	<i>maximum likelihood</i>
NLS	<i>nonlinear least square</i>
RADAR	<i>radio detection and ranging</i>
RANSAC	<i>random sampling consensus</i>
ROV	<i>remotely operated vehicle</i>
SLAM	<i>simultaneous localisation and mapping</i>
SFM	<i>structure from motion</i>
SIFT	<i>scale-invariant feature transform</i>
SONAR	<i>sound navigation and ranging</i>
UAV	<i>unmanned aerial vehicle</i>
UUV	<i>unmanned underwater vehicle</i>

SYMBOLS AND OPERATORS

Notation	Meaning
x_t	State vector
$x_{0:t}$	State trajectory up to time t
y_t	Measurement vector
$y_{1:N}$	Data batch of length N also denoted $y_{1:N} = Y = \{y_1, \dots, y_N\}$
u_t	Input vector
θ	Parameter vector
$\hat{\theta}$	Estimate of θ
e_t	Measurement noise
w_t	Process noise noise
Q	Process noise variance
R	Measurement variance
$f(x_t)$	Process model
$h(x_t)$	Measurement model
$\mathcal{N}(\mu, \Sigma)$	Gaussian distribution with mean μ and variance Σ
$p(Y \theta), p_\theta(Y)$	Data likelihood
$p(\theta Y)$	Posterior density
R	Rotation matrix
ω	Angular velocity
p	Position
v	Velocity
a	Acceleration
R^T	Transpose of matrix or vector
q	Unit quaternion $q = [q_0, q_1, q_3, q_4]^T$
$\det$	Determinant
$\times$	Cross product
∇	Gradient operator
$\mathbb{R}$	Set of real numbers
$P([X, Y, Z]^T)$	Projection operator $P([X, Y, Z]^T) = [X/Z, Y/Z]^T$
$\frac{\ x\ _{P^{-1}}}{\sqrt{x^T P^{-1} x}}$	P -weighted norm of vector x
$SO(3)$	Special Orthogonal Group
E	Essential Matrix
$\arg \max_{\theta}$	Maximising argument with respect to θ
$\arg \min_{\theta}$	Minimising argument with respect to θ
$\sim$	Distributed according to. Or up to an unknown scale λ according to $a \sim b \iff a = \lambda b$.

Part I

Background

1

Introduction

This thesis is about navigation, mapping and modeling for mobile robotics. The specific applications studied are sensor network localisation, underwater vehicle modeling and batched *simultaneous localisation and mapping* (SLAM) initialisation and estimation methods for inertial sensors and monocular cameras.

This chapter introduces and motivates the research topics in this thesis. It also summarises the research contributions by the author of the appended publications and other publications.

1.1 Motivation

Autonomous systems are becoming increasingly important in today's society. The main drive for industrial automatisisation is increased efficiency leading to a larger competitive advantage. In our homes we are enjoying the benefits of many automatised processes such as dishwashers, heat exchangers and robotic lawnmowers. We use these systems because it allows us to do things that are more rewarding, such as building autonomous systems. The design, construction and evaluation of these systems is an iterative process which may involve machines, computers, physical models, mathematical models and a fair bit of experience.

Today, autonomous systems are not only restricted to industries and our homes but are becoming integrated into our normal day lives. For instance, yesterday's fantasies about self-driving cars are now becoming reality¹. Autonomous robotics see, Figure 1.1, is a special branch in mobile robotics in which the focus is to automatise as many subsystems of the robot as possible. A research-intensive

¹<http://googleblog.blogspot.se/2014/04/the-latest-chapter-for-self-driving-car.html>

Figure 1.1: Parts of autonomous robotic systems.

Figure 1.2: Left: Combined camera and inertial sensors in a single unit. Prototype made by XSens. Right: The Saab V-200 Skeldar helicopter. By courtesy of Saab AB.

part of this subsystem is onboard sensor data interpretation which is a cornerstone for making truly autonomous robots without the need for external support systems. The processed sensor data is used for various purposes such as navigation, mapping, control and exploration.

SLAM emerged in the field of mobile robotics as a tool to enhance navigation and to infer properties of the surrounding environment by means of the onboard sensors. As the name suggests, SLAM is the joint problem of localisation and mapping which, on their own, are active research fields. In Figure 1.2 a multisensor unit and an unmanned autonomous vehicle are shown.

Localisation, or navigation, problems assume that the surrounding environment is, to some extent, known a priori. Such information can for instance be; topographic maps, radio beacons, star constellations, magnetic fields, *global naviga-*

Figure 1.3: Left: The Saab Seaeye Falcon ROV. By courtesy of Saab AB. Right: Position trajectory based on fusion of the ROV hydrodynamic model and magnetometer in magenta and the cyan curve marks the cable which the ROV roughly followed.

tion satellite systems (GNSS) satellite ephemeris, which are integrated directly on moving platforms. If localisation of moving objects, i.e., tracking, is done using external systems such as; ground based radar, closed-circuit television, cellular networks, and other types of sensor networks, the location of the sensor nodes needs to be accurately known. In Figure 1.3 a navigation example using a *remotely operated vehicle* (ROV) is shown.

Mapping problems consider estimation of the surrounding environment with examples such as; topographic maps, bathymetric maps, magnetic fields, object shape. It is assumed that all necessary information about the moving platforms is available such that the uncertainty w.r.t., to the estimated maps is negligible. This also covers cases when moving platforms are measured by sensor networks and the node locations need to be calibrated. In Figure 1.4 a small experimental setup and the corresponding SLAM estimate are shown.

SLAM research initially spent a lot of effort on indoor platforms typically equipped with wheel encoders for odometry and line sweeping laser range scanners and these are still common. Today, this field of research is huge and heterogeneous with all kinds of platform and sensor combinations. A small table with some sensors and platforms that have been used in SLAM applications are shown in Table 1.1.

1.2 Contributions

Published and submitted publications which constitute the second part of this thesis are listed below in chronological order together with a description and the contribution of each paper.

Figure 1.4: Left: Image of an environment used for SLAM experiments using camera and inertial sensors. Right: SLAM results using EKF and (smoothed) nonlinear least-squares.

Sensor\Platform	Flying	Ground	Pedestrian	Vessel	Submarine
Camera	1	2	3	4	5
Stereo Camera	6	7	8		9
RADAR	10	11		12	
SONAR		13		14	15
IMU	16	17	18	19	20
LiDAR	21	22		23	

Table 1.1: Some SLAM publications using various platforms and sensors.

[1](#) (Karlsson et al., 2008; Caballero et al., 2009; Bryson and Sukkarieh, 2009; Lupton and Sukkarieh, 2009, 2008; Bryson and Sukkarieh, 2008), [2](#) (Wang and Dissanayake, 2012; Thrun and Montemerlo, 2006; Callmer et al., 2008), [3](#) (Davison et al., 2007; Davison, 2003; Eade, 2008; Klein and Murray, 2007), [4](#) (Callmer et al., 2011), [5](#) (Kim, 2013), [6](#) (Jung and Lacroix, 2003), [7](#) (Konolige and Agrawal, 2008; Cummins and Newman, 2010), [8](#) (Lupton and Sukkarieh, 2012; Karlsson and Bjärkefur, 2010; Strasdat et al., 2011; Jung and Taylor, 2001), [9](#) (Eustice et al., 2006; Mahon et al., 2008), [10](#) (Sjanic and Gustafsson, 2010), [11](#) (Marck et al.; Gerossier et al., 2009), - (), [12](#) (Callmer et al., 2011; Mullane et al., 2010), [13](#) (Choi et al., 2005), - (), [14](#) (Wrobel, 2014), [15](#) (Newman et al., 2003; Ribas et al., 2006), [16](#) (Bryson et al., 2010), [17](#) (Wijesoma et al., 2006), [18](#) (Lupton and Sukkarieh, 2012), [19](#) (Han and Kim, 2013), [20](#) (Kim and Eustice, 2009), [21](#) (Fossel et al., 2013), [22](#) (Bosse and Zlot, 2008), [23](#) (Han and Kim, 2013).

1.2.1 Included Contributions

Silent Localization of Underwater Sensors Using Magnetometers

J. Callmer, M. Skoglund, and F. Gustafsson. Silent localization of underwater sensors using magnetometers. *EURASIP Journal on Advances in Signal Processing*, 2010, 2010. doi: 10.1155/2010/709318. URL <http://dx.doi.org/10.1155/2010/709318>. Article ID 709318.

Paper A presents a method for localisation of underwater sensors equipped with triaxial magnetometers using a friendly vessel with known magnetic characteristics.

Underwater sensor networks can be used to detect and track surface vessels but more importantly, underwater vessels which may pose threats in security critical maritime environments. For these networks to function properly the location of each sensor need to be accurately known which can be difficult to obtain in fast deployment scenarios. We here demonstrate how the sensor positions and the vessel trajectory can be estimated simultaneously in an *extended Kalman filter* (EKF) using only magnetometer measurements from the sensors. *Cramér-Rao lower bound* (CRLB) analysis shows the attainable node localisation accuracy for a given trajectory and the attainable tracking performance for a given network. The CRLB analysis could thus serve as a guide on sensor network design. Sensitivity analysis indicates that when using GNSS measurement of the vessel trajectory, errors in sensor orientation and magnetic dipole strength are suppressed and the localisation accuracy is enhanced.

The results are evaluated using simulated data. This is also the only publication in this thesis without real, experimental, data. This inexpensive solution to the difficult sensor localisation problem may also be used for re-localisation of sensors if conditions have changed.

This is joint work primarily between the first and the second author who produced the ideas, theory, implementation and most of the writing.

A Nonlinear Least Squares Approach to the SLAM Problem

Z. Sjanic, M. A. Skoglund, F. Gustafsson, and T. B. Schön. A nonlinear least squares approach to the SLAM problem. In *Proceedings of the IFAC World Congress*, volume 18, Milan, Italy, 28-2 Aug./Sept. 2011.

Paper B presents an algorithm for visual/inertial SLAM based on the *maximum a posterior* (MAP) estimate of the whole 6 *degrees-of-freedom* (DOF) trajectory (including velocities) and 3D map, solved using *nonlinear least-squares* (NLS) optimisation.

An initial estimate is acquired using EKF-SLAM with inertial data as input to the motion model. *Scale invariant feature transform* (SIFT) image features are used for tracking and data association and the *inverse depth parametrisation* (IDP) is used for landmark parametrisation since it is known to handle depth uncertainty, due to EKF linearisation errors, better than using the direct 3D parametrisation.

The inherent sparsity structure of the full SLAM problem is efficiently utilised in the NLS solver and the result is a metrically correct position, orientation, velocity and landmark map estimate. The proposed algorithm is evaluated on experimental data using a sensor platform mounted on an industrial robot enabling accurate ground truth reference.

The results from this algorithm can, for instance, be used on unmanned aerial vehicles to compute detailed terrain maps necessary for safe landing or in underwater localisation and mapping and other GNSS denied environments.

This is joint work primarily between the first and the second author who produced the ideas, theory, experiments, implementation and most of the writing.

Parts of this work has also been used in Nilsson (2010); Nilsson et al. (2011) where the sensor is a forward looking *infrared* (IR) camera attached to a car, and in Karlsson and Bjärkefur (2010) where the authors use stereo camera and laser camera in an indoor environment.

Modeling and Sensor Fusion of a Remotely Operated Underwater Vehicle

M. A. Skoglund, K. Jönsson, and F. Gustafsson. Modeling and sensor fusion of a remotely operated underwater vehicle. In *Proceedings of the 15th International Conference on Information Fusion (FUSION)*, Singapore, 9-12 July 2012, pages 947–954. IEEE, 2012.

Paper C presents how a complex hydrodynamic model for a *remotely operated vehicle* (ROV) can be used to robustify navigation based on onboard sensors.

ROV's are small and relatively cheap *unmanned underwater vehicles* (UUV). They are used in situations such as: tunnel or hull inspections; deep sea missions beyond human diver depths; mine hunting and mine disposal. Due to their limited payload capacity, and lack of external underwater localisation support, the onboard navigation systems have to be robust. The paper shows how previously estimated model structures are put together in a large model describing the full 6-DOF motion of the ROV, including angular and linear velocities. The model potentially provides an independent source of vehicle speed and angular rate. We also provide models for all the onboard sensors, expressed in the body frame of the ROV, which are fused with the hydrodynamic model in an EKF. The results are based on data from the field tests performed in the Master's thesis work of Jönsson (2010) which the first author supervised. We show that, in particular, the vehicle speed can be accurately predicted as compared with the doppler speedometer.

This is joint work primarily between the first and the second author. The second author provided initial models and experiments. The first author did most of the implementation and most the writing.

Initialisation and Estimation Methods for Batch Optimisation of Inertial/Visual SLAM

M. A. Skoglund, Z. Sjanic, and F. Gustafsson. Initialisation and estimation methods for batch optimisation of inertial/visual SLAM. Submitted to IEEE Transactions on Aerospace and Electronic Systems, June 2014.

Paper D presents a complete initialisation method for inertial/visual batch SLAM which can be used to obtain metrically correct map and navigation state estimates.

Batch formulations of inertial/visual SLAM are nonlinear and nonconvex problems which need a good initial estimate to converge to the global optimum. In Paper B the full SLAM problem was initialised using EKF-SLAM which is not a feasible solution for large data sets. In this paper we present a multi-step algorithm that solves a series of almost uncoupled problems. The combination of rotation estimation and appearance based data association using only vision together with visual/inertial methods leads to almost linear formulations which can be solved easily. The initialisation method is demonstrated on both simulated data and a small feasibility study on experimental data using an industrial robot, to get access to ground truth, is also performed.

This is joint work primarily between the first and the second author who produced the ideas, theory, experiments, implementation and most of the writing.

EM-SLAM with Inertial/Visual Applications

Z. Sjanic, M. A. Skoglund, and F. Gustafsson. EM-SLAM with inertial/visual applications. Submitted to IEEE Transactions on Aerospace and Electronic Systems, June 2014.

Paper E presents an *expectation-maximisation* (EM) approach to a *maximum likelihood* (ML) batch formulation of inertial/visual SLAM.

The EM algorithm introduces a set of so-called latent, or hidden, variables. By doing so, the problem can be split into two, hopefully, simpler problems. The first problem is the expectation, the so-called E-step, with respect to the conditional density of the hidden variables. The second problem, known as the M-step, is to maximise the result obtained in the E-step with respect to the unknown parameters. These two problems are solved iteratively until some convergence criterion is met. The EM-SLAM algorithm proposed here considers the platform motion as hidden variables and the landmark map as the unknown parameters. The E-step is solved using an *extended Rauch-Tung-Striebel smoother* (E-RTS) with constant size state vector which becomes computationally cheap. The M-step is solved efficiently with a quasi-Newton method having fewer variables than the full NLS formulation in which both the state sequence and the map are seen as parameters in an ML fashion. EM-SLAM is compared with NLS-SLAM both in terms of performance and complexity. The proposed method is evaluated in real experiments and also in simulations on a platform with a monocular cam-

era attached to an inertial measurement unit. The initial estimate for both the parameters and the hidden variables are obtained using the method in Paper D. It is demonstrated to produce lower *root mean square error* (RMSE) than with a standard Levenberg-Marquardt solver of NLS problem, at a computational cost that increases considerably slower.

This is joint work primarily between the first and the second author who produced the ideas, theory, experiments, implementation and most of the writing.

1.2.2 Additional Publications

This section contains published work not included in this thesis.

1.2.3 Optical See-Through Head Mounted Display Calibration

Augmented reality (AR) is used to denote visual information overlaid, augmented, on human vision. *Optical see-through head mounted displays* (OSTHMD) are wearable AR systems, which means that the OSTHMD displayed graphics moves with the user, see Figure 1.5. To augment the real world with graphics in such a system three main problems need to be solved. First, the pose (position and orientation) of the OSTHMD with respect to the inertial frame needs to be known. Second, the calibration problem which constitutes the static transformations of the relative position and rotation of the semitransparent graphics screen with respect to the user's eye and an inertial frame. Finally, models of the environment where the graphics should be superimposed have to be created. In the two articles, the first two of the three problems are addressed. The pose of the user's head, and consequently the OSTHMD, is provided using a visual tracking system along with a visual landmark coordinate. The calibration problem is solved using the data from the tracking system and measurements acquired by the subject through a bore-sighting exercise. The user aligns graphics (a cross-hair) displayed on the screen with a measured point (a diode) in the inertial frame, see Figure 1.5, and several such alignments are collected from different screen points and the subject's locations in order to excite the parameter space.

We adopt the theoretical framework for camera calibration founded in the computer vision and photogrammetry domains to OSTHMD calibration. The calibration problem itself is rather ill-posed since the measurements are few compared to the parameter space and the *signal-to-noise ratio* (SNR) is low. The work in these two publications reflects some labour intense engineering/research where several months were spent on the AR system which consists of several hardware components interacting through several software layers.

This is joint work primarily between the first and the second author who produced the ideas, theory, experiments and implementation. While the first author did most of the writing.

Figure 1.5: Left: A Kaiser ProView 50ST Optical see-through head mounted display equipped with IR diodes for the PhaseSpace IMPULSE motion tracking system placed on a mannequin. Right: Example of a bore-sighting exercise with the reindeer-looking OSTHMD system. Note the diode in the far back of the room which the subject (myself) aims at.

Optical See-Through Head Mounted Display Direct Linear Transformation Calibration Robustness in the Presence of User Alignment Noise

M. Axholt, M. Skoglund, S. D. Peterson, M. D. Cooper, T. B. Schön, F. Gustafsson, A. Ynnerman, and S. R. Ellis. Optical see-through head mounted display direct linear transformation calibration robustness in the presence of user alignment noise. In *Proceedings of the 54th Annual Meeting of the Human Factors and Ergonomics Society*, volume 54, pages 2427–2431, San Francisco, CA, USA, 27-1 Sept./Oct. 2010. Human Factors and Ergonomics Society.

The abstract from the paper is included below.

The correct spatial registration between virtual and real objects in optical see-through augmented reality implies accurate estimates of the user's eyepoint relative to the location and orientation of the display surface. A common approach is to estimate the display parameters through a calibration procedure involving a subjective alignment exercise. Human postural sway and targeting precision contribute to imprecise alignments, which in turn adversely affect the display parameter estimation resulting in registration errors between virtual and real objects. The technique commonly used has its origin in computer vision, and calibrates stationary cameras using hundreds of correspondence points collected instantaneously in one video frame where precision is limited only by pixel quantization and image blur. Subsequently the input noise level is several order of magnitudes greater when a human operator manually collects correspondence points one by one. This paper investigates the effect of human alignment noise on view parameter estimation in an optical see-through head mounted display to determine how well a standard camera calibration method performs at greater noise levels than documented in computer vision literature. Through Monte-Carlo simulations we show that it is particularly difficult to estimate the user's eyepoint in depth, but

that a greater distribution of correspondence points in depth help mitigate the effects of human alignment noise

Parameter Estimation Variance of the Single Point Active Alignment Method in Optical See-Through Head Mounted Display Calibration

M. Axholt, M. A. Skoglund, S. D. O'Connell, M. D. Cooper, S. R. Ellis, and A. Ynnerman. Parameter estimation variance of the single point active alignment method in optical see-through head mounted display calibration. In *Proceedings of the IEEE Virtual Reality Conference*, pages 27–34, Singapore, Republic of Singapore, Mar. 2011.

The abstract from the paper is included below.

The parameter estimation variance of the *single point active alignment method* (SPAAM) is studied through an experiment where 11 subjects are instructed to create alignments using an *optical see-through head mounted display* (OSTHMD) such that three separate correspondence point distributions are acquired. Modeling the OSTHMD and the subject's dominant eye as a pinhole camera, findings show that a correspondence point distribution well distributed along the user's line of sight yields less variant parameter estimates. The estimated eye point location is studied in particular detail. The findings of the experiment are complemented with simulated data which show that image plane orientation is sensitive to the number of correspondence points. The simulated data also illustrates some interesting properties on the numerical stability of the calibration problem as a function of alignment noise, number of correspondence points, and correspondence point distribution.

Insights from Implementing a System for Peer-Review

C. Lundquist, M. A. Skoglund, K. Granström, and T. Glad. Insights from implementing a system for peer-review. *IEEE Transactions on Education*, 56(3):261–267, 2013.

Finally, the following paper about undergraduate teaching and peer-review has been published.

1.3 Thesis Outline

The thesis is divided into two parts, with background material in the first part and with edited versions of published papers in the second part. The first part consists of material introducing and explaining the background to the publications. The publications on their own include detailed background material.

The first part of the thesis is organised as follows. Chapter 2 presents model structures for motion, sensors and computer vision. Chapter 3 introduces the sensor fusion concept which is used for both filtering and smoothing. The connection to optimisation is also explained. Chapter 4 gives a brief overview of SLAM estimation methods and explains the concepts used in the publications. The first

part of the thesis ends with Chapter 5, which presents conclusions and discusses future work.

The second part of the thesis consist of edited versions of the papers. The papers are only edited to comply with the layout of the thesis template.

2

Models

In this chapter the most important model structures which are used in the publications in Part II are outlined. The models describe rigid body kinematics and dynamics, the pinhole camera, inertial measurements, magnetometers and three different landmark parametrisations for camera measurement models. This is the basis for inference in state-space, and other parametric, models which are described in the next chapter.

In a navigation context the state vector may include:

- Position $\mathbf{p}$, velocity $\mathbf{v}$, acceleration $\mathbf{a}$ and possibly higher order derivatives.
- Unit quaternion q parametrising orientation, angular velocity ω and higher order derivatives.

For localisation and tracking problems the state vector may include:

- Stationary point targets as landmark coordinates m .
- Stationary sensors $\mathbf{p}$ which may also include sensor orientation q and other quantities.
- Non-stationary targets possibly constrained to different motion models.
- Extended targets, stationary or non-stationary.
- Binary correspondence variables c relating measurements to target identities and possibly other metadata.

In case navigation and localisation are estimated jointly it is natural to include the appropriate quantities mentioned above in a large state vector.

2.1 Kinematics

Kinematics describes the motion of bodies without considering the forces causing the motion. Kinematic transformations between coordinate frames consist of length preserving translations and rotations as illustrated in Figure 2.1.

Figure 2.1: Rigid body transformations.

2.1.1 Translational Kinematics

A translation is a displacement of the body origin p while the axes are kept aligned. The translation is a vector, p^w , from the origin of the frame w to the coordinate p expressed in the w -frame and it is also called the position. The translational motion equations can be derived from the time-derivative of a translation

$$\ddot{p} = \dot{v} = a, \quad (2.1)$$

where v is the velocity and a is the acceleration. It is of course possible, and sometimes necessary, to introduce higher order terms such as $\dot{a}$ which is known as jerk, but most often it can be considered to be noise i.e., $\dot{a} = w_a$. In (2.1) there is no indication of the dimension in which the motion occurs. This is however no restriction since the motion is independent in each dimension. The corresponding matrix form is

$$\begin{bmatrix} \dot{p} \\ \dot{v} \\ \dot{a} \end{bmatrix} = \begin{bmatrix} 0 & I & 0 \\ 0 & 0 & I \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} p \\ v \\ a \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ I \end{bmatrix} w_a, \quad (2.2a)$$

$$\Longleftrightarrow \dot{x} = Ax + Bw_a, \quad (2.2b)$$

where the matrices 0 and I are of appropriate dimensions. Assuming that the input w_a is constant over the sampling interval, T , the ordinary differential equa-

tion (2.2) is solved as

$$x_{t+1} = e^{AT} x_t + \int_0^T e^{A\tau} d\tau B w_a, \quad (2.3)$$

which results in

$$\begin{bmatrix} p_{t+1} \\ v_{t+1} \\ a_{t+1} \end{bmatrix} = \begin{bmatrix} I & TI & \frac{T^2}{2}I \\ 0 & I & \frac{T}{1}I \\ 0 & 0 & I \end{bmatrix} \begin{bmatrix} p_t \\ v_t \\ a_t \end{bmatrix} + \begin{bmatrix} \frac{T^3}{6}I \\ \frac{T^2}{2}I \\ I \end{bmatrix} w_a. \quad (2.4)$$

2.1.2 Rotational Kinematics

Rotational kinematics is more complicated than its translational counterpart and this is due to the fact that rotation representations are nonlinear. This means that some ordinary operations defined on vector spaces, such as addition, are not defined here. Proper rotations are linear maps that preserve length and the handedness of the basis. There are many parametrisation alternatives to rotation matrices and perhaps the most common ones are Euler angles and the unit quaternion. For a thorough overview of rotation parameterisations see e.g., Shuster (1993). In this thesis, rotation in 3D is the typical case considered and as a common denominator for all parameterisations is that they can be transformed into a corresponding rotation matrix $R \in SO(3)$.

Definition 2.1 (Special Orthogonal Group $SO(3)$). A matrix $R \in \mathbb{R}^{3 \times 3}$ belongs to the special orthogonal group $SO(3)$ if and only if it holds

$$R^T R = I, \quad (2.5a)$$

$$\det R = 1. \quad (2.5b)$$

Figure 2.1a describes a rotation from the w frame to the c frame. This can be parametrised by the rotation matrix R^{cw} . The inverse rotation direction using (2.5a) is $(R^{cw})^{-1} = (R^{cw})^T$, hence $(R^{cw})^T = R^{wc}$.

Spatial rotations intuitively only have three degrees of freedom yet they still need at least five parameters (Hopf, 1940) in order to represent a global description see Stuelpnagel (1964) for a discussion. Rotations matrices are impractical for several reasons e.g., when rotations are estimated since it is difficult to design estimators directly on $SO(3)$ and it is costly to use nine parameters. Luckily there are many alternatives for rotation parametrisation and an appealing option is the unit quaternion. Quaternions were invented by Hamilton (1844) as a tool to extend the imaginary numbers. The unit length quaternion is widely used in aerospace industry, mechanics, computer graphics, among others, since it allows a computationally efficient and singularity free rotation representation. The unit quaternion is a vector of real numbers $q \in \mathbb{R}^4$. A definition is given in Kuipers (2002, page 104), who splits the vector a scalar part, q_0 , and a vector, $q = e_1 q_1 +$

$e_2q_2 + e_3q_3$, where e_1, e_2, e_3 is the standard orthonormal basis in $\mathbb{R}^3$. With Kuipers definition the full vector representation is

$$q = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} = \begin{bmatrix} q_0 \\ \mathbf{q} \end{bmatrix}. \quad (2.6)$$

Quaternions can be used to represent a rotation in $\mathbb{R}^3$ but in order to do so they must be constrained to the unit sphere $q \in S^3$, i.e., $q^T q = 1$, hence the name unit quaternion. Let

$$q = \begin{bmatrix} \cos \delta \\ \sin \delta \mathbf{n} \end{bmatrix} \quad (2.7)$$

which is a unit quaternion describing a rotation of an angle 2δ around the unit vector $\mathbf{n} \in \mathbb{R}^3$. Then a rotation using a unit quaternion is

$$\tilde{x}^b = q^{ba} \odot \tilde{x}^a \odot q^{ab} \quad (2.8)$$

where $\tilde{x}^* = [0, x^*]$ are the vectors' quaternion equivalents and $\odot$ denotes the quaternion multiplication, see e.g., Törnqvist (2008); Hol (2011). The unit quaternion can also be used to construct a rotation matrix, see for instance Shuster (1993, page 462), where the superscript ab is omitted for the sake of readability,

$$R(q) = \begin{bmatrix} (q_0^2 + q_1^2 - q_2^2 - q_3^2) & 2(q_1q_2 + q_0q_3) & 2(q_1q_3 - q_0q_2) \\ 2(q_1q_2 - q_0q_3) & (q_0^2 - q_1^2 + q_2^2 - q_3^2) & 2(q_2q_3 + q_0q_1) \\ 2(q_1q_3 + q_0q_2) & 2(q_2q_3 - q_0q_1) & (q_0^2 - q_1^2 - q_2^2 + q_3^2) \end{bmatrix} = \quad (2.9)$$

$$= \begin{bmatrix} (2q_0^2 + 2q_1^2 - 1) & 2(q_1q_2 + q_0q_3) & 2(q_1q_3 - q_0q_2) \\ 2(q_1q_2 - q_0q_3) & (2q_0^2 + 2q_2^2 - 1) & 2(q_2q_3 + q_0q_1) \\ 2(q_1q_3 + q_0q_2) & 2(q_2q_3 - q_0q_1) & (2q_0^2 + 2q_3^2 - 1) \end{bmatrix}, \quad (2.10)$$

where the last equality is established using $q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1$. It should be noted that the unit quaternion is not a global parametrisation because the q and $-q$ describe the same rotation i.e., $R(q) = R(-q)$ which is easily verified from (2.9). This is seldom a practical problem since a solution with e.g., positive q_0 component can be chosen.

A local parametrisation which is popular within the computer vision community is the so-called exponential coordinates, see e.g., Ma et al. (2003). It uses a three-dimensional vector $\omega \in \mathbb{R}^3$ such that for any $R \in SO(3)$ there exists $R = e^{[\omega]_\times}$ where

$$[\omega]_\times = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} \quad (2.11)$$

is the matrix form of the cross product. It should be noted that any vector of the form $\omega + 2\pi k\omega$, where k being an integer, have the same rotation matrix which also means that for a given rotation matrix there are infinitely many exponential coordinates. The rotation matrix can be efficiently computed from the

exponential parameters using Rodrigues' formula

$$R = e^{[\omega]_{\times}} = I + \frac{[\omega]_{\times}}{\|\omega\|} \sin(\|\omega\|) + \frac{[\omega]_{\times}^2}{\|\omega\|^2} (1 - \cos(\|\omega\|)), \quad (2.12)$$

which follows from the Taylor expansion of the exponential function, see Ma et al. (2003, Theorem 2.9) for a proof.

A differential equation for rotations can be established by using some properties of rotation matrices. Using the definition of $SO(3)$ and taking the time derivative of both sides of (2.5a) as

$$\frac{d}{dt}(RR^T) = \frac{d}{dt}I, \quad \Longleftrightarrow \quad (2.13)$$

$$\dot{R}R^T + R\dot{R}^T = 0, \quad (2.14)$$

which implies that the matrix products are skew-symmetric

$$\dot{R}R^T = -(\dot{R}R^T)^T. \quad (2.15)$$

We can thus define the right hand side as

$$\dot{R}R^T = [\omega]_{\times}, \quad (2.16)$$

and since $R^TR = I$ we have that

$$\dot{R} = [\omega]_{\times} R. \quad (2.17)$$

By solving the differential equation (2.17) with ω constant

$$R(t) = e^{[\omega]_{\times}t} R(0). \quad (2.18)$$

the exponential coordinates are obtained. Similarly, the differential form of the unit quaternion parametrisation is given by

$$\dot{q}_t = \underbrace{\frac{1}{2} \begin{bmatrix} 0 & -\omega_x & -\omega_y & -\omega_z \\ \omega_x & 0 & \omega_z & -\omega_y \\ \omega_y & -\omega_z & 0 & \omega_x \\ \omega_z & \omega_y & -\omega_x & 0 \end{bmatrix}}_{S(\omega)} \underbrace{\begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix}}_{\tilde{S}(q)} = \frac{1}{2} \underbrace{\begin{bmatrix} -q_1 & -q_2 & -q_3 \\ q_0 & -q_3 & q_2 \\ q_3 & q_0 & -q_1 \\ -q_2 & q_1 & q_0 \end{bmatrix}}_{\tilde{S}(q)} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}, \quad (2.19)$$

which has the nice property of being bilinear. Note that due to the algebraic constraint the differential form of the unit quaternion is a *differential algebraic equation* (DAE). For a complete derivation of these relations see e.g., Törnqvist (2008); Hol (2011). The unit quaternion is thus solved as

$$\dot{q} = \frac{1}{2} S(\omega) q \implies \quad (2.20)$$

$$q(t) = e^{\frac{1}{2} S(\omega)t} q(0) \quad (2.21)$$

where it is assumed that the angular velocity is constant. Defining the skew-symmetric matrix $S = \frac{1}{2} S(\omega)t$, $q_1 = q(t)$ and $q_0 = q(0)$ the quaternion length

remains unchanged

$$\|q_1\|_2^2 = \|e^S q_0\|_2^2 = q_0^T (e^S)^T e^S q_0 = q_0^T e^{S^T + S} q_0 = q_0^T q_0 = \|q_0\|_2^2, \quad (2.22)$$

since $S = -S^T$. The matrix exponential of the orientation dynamics can further be simplified

$$q_t = e^{\frac{1}{2}S(\omega)T_s} q_0 = \left(\cos\left(\frac{\|\omega\|T_s}{2}\right) I + \frac{\sin\left(\frac{\|\omega\|T_s}{2}\right)}{\|\omega\|} S(\omega) \right) q_0 \approx \left(I + \frac{T_s}{2} S(\omega) \right) q_0 \quad (2.23)$$

where the approximation is based on the small angle approximation which is good if the sampling rate T_s is high. It is now simple to introduce noise and bias as

$$q_t = \left(I + \frac{T_s}{2} S(\omega + \mathbf{b}_\omega + w_\omega) \right) q_0 = \left(I + \frac{T_s}{2} S(\omega + \mathbf{b}_\omega) \right) q_0 + \frac{T_s}{2} \widetilde{S}(q_0) w_\omega \quad (2.24)$$

using the relation (2.19). It is important to note that the nonlinear constraint $q^T q = 1$ need to be handled correctly in any estimator. In the case of filtering there are mainly two options. The first is to introduce a fictitious measurement $y = 1 - q^T q$ which should be zero and the second option is to normalise the updated quaternion as $q := q/\|q\|$ which is a projection onto the unit sphere.

Using unit quaternions for filtering may turn out problematic since the process noise covariance and the true state covariance is singular due to the over-parametrisation. Also, if a smoothed estimate is sought problems are unavoidable using unconstrained formulations, see Shuster (2003). Thus, there is room for improving the smoothing approach using unit quaternions as in Paper B.

A practical option when quaternions are simulated in continuous time is to introduce a feedback

$$\dot{q} = \frac{1}{2} S(\omega) q + \frac{\gamma}{2} (1 - q^T q) q, \quad (2.25)$$

which drives the quaternion to unit length, here γ is a proportional positive gain. This approach was used in the hydrodynamic model in Paper C.

2.1.3 Rigid Body Kinematics

Combining translational and rotational kinematics can be done in several ways and is very much a design task that depends on the system at hand. A straightforward model is given by the combination of the constant acceleration model and (2.2) and the unit quaternion (2.19) as

$$\begin{bmatrix} \dot{\mathbf{p}} \\ \dot{\mathbf{v}} \\ \dot{\mathbf{a}} \\ \dot{q} \end{bmatrix} = \begin{bmatrix} 0 & I & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{2} S(\omega) \end{bmatrix} \begin{bmatrix} \mathbf{p} \\ \mathbf{v} \\ \mathbf{a} \\ q \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ I \\ 0 \end{bmatrix} w_a, \quad (2.26)$$

and it is also quite common to extend the model with states for angular acceleration, and possibly bias states for gyroscopes and accelerometers and this is done in Paper D.

Another model is obtained by considering acceleration and angular rates to be input to the system

$$\begin{bmatrix} \dot{\mathbf{p}} \\ \dot{\mathbf{v}} \\ \dot{\mathbf{q}} \end{bmatrix} = \begin{bmatrix} 0 & I & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{p} \\ \mathbf{v} \\ \mathbf{q} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ I & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \mathbf{a} + \mathbf{b}_a \\ \frac{1}{2} S(\boldsymbol{\omega} + \mathbf{b}_\omega) \mathbf{q} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ I & 0 \\ 0 & \frac{1}{2} \widetilde{S}(\mathbf{q}) \end{bmatrix} \begin{bmatrix} \mathbf{w}_a \\ \mathbf{w}_\omega \end{bmatrix}. \quad (2.27)$$

This significantly reduces the state space at the cost of limited noise modeling and fault detection capabilities. It should be noted that the biases are not observable using only accelerometers and gyroscopes in here, but must be inferred from other observations, e.g., camera measurements. Models with inertial sensors as input signals are used in Paper B and Paper E. In Paper A the kinematics of a vessel is described by a coordinated turn model which essentially constrains the object to follow a circular path. Such models are rather flexible and may be used to simulate more advanced models. It may be convenient to express some parts of a motion model in the body fixed frame since forces and torques are often naturally represented in this frame. This is the case in Paper C when modelling of a ROV is considered since the expressions for linear and rotational velocities due to external forces become much simpler.

2.2 Rigid Body Dynamics

Rigid body dynamics in classical mechanics investigates the motion of objects as caused by forces and torques. The fundamental equations describing these relations are given by Euler's axioms of motion. The first axiom is

$$\left(\frac{d(m\dot{\mathbf{p}})}{dt} \right)_i = \sum_{j=1}^N F_j, \quad (2.28)$$

which gives the relation of a body's acceleration $\ddot{\mathbf{p}}$ in an inertial frame i with mass m which is due to external forces F_j and $\mathbf{p}$ is the centre of mass. This is the straightforward extension of Newton's second law of motion for particles to rigid bodies expressed as conservation of linear momentum. The corresponding law for angular momentum, known as Euler's second law, states that the change of angular momentum of the body is equal to all external torques about the origin

$$\left(\frac{d(J\boldsymbol{\omega})}{dt} \right)_i = \sum_{j=1}^N \mathbf{r}_j \times F_j, \quad (2.29)$$

where J is the inertia matrix, $\boldsymbol{\omega}$ is the angular velocity of the rigid body, $\mathbf{r}_j$ are vectors that point from the center of rotation to the points where the external forces F_j are applied. It is useful to express Euler's laws in body referenced velocities since the mass and inertia are then constant. This can be done considering the derivative of vectors in a rotating reference frame, b , using the relation

$$\left(\frac{d\mathbf{r}}{dt} \right)_i = \left(\frac{d\mathbf{r}}{dt} \right)_b + \boldsymbol{\omega} \times \mathbf{r}. \quad (2.30)$$

in which the body referenced derivative is the same if the inertial frame is not rotating, i.e., $\omega = 0$. Applying (2.30) to (2.28) results in

$$m\dot{p}^b = \sum_{j=1}^N F_j - \omega \times m\dot{p}^b, \quad (2.31)$$

where it is assumed that the mass is constant. Similarly, the expression for angular velocity (2.29) is

$$J\dot{\omega}^b = \sum_{j=1}^N r_j \times F_j - \omega^b \times J\omega^b. \quad (2.32)$$

For a complete derivation, see e.g., Fossen (2011).

2.3 Inertial Sensors

Inertial sensors measure specific force and angular velocity. These have great importance for many navigation systems since they provide an independent source for computing the position and orientation of a moving platform relative to some initial pose by integrating the accelerations twice and the angular velocity once, respectively. It may sometimes be the only means of navigation when other systems are unavailable. For instance, GNSS are prohibitive in many places such as underwater environments or indoors, and are also subject to jamming and spoofing. Magnetometers are passive sensors that can be used to calculate the orientation w.r.t., the local magnetic field. However, this field is rather weak and easily disturbed by ferrous objects. Inertial sensors are not subject to such external errors but rather errors due to the internal workings of the sensors. These contribute to the integration drift which means that the true pose of the sensor will deteriorate when the measurements are integrated. This drift is roughly inversely proportional to the price of the sensors. In this work we will only consider *micro electrical mechanical systems* (MEMS) type IMUs due to their relatively low cost and small size. There is a rich body of literature on navigation in general and modeling of IMU's in particular, see for instance Titterton and Weston (1997); Britting (1971). In all applications here the IMU's are rigidly mounted to the moving platform, in a so-called strap-down configuration, and hence sensor readings are naturally referenced in this frame.

2.3.1 Gyroscopes

Commonly, MEMS gyroscopes are based on measuring the rotation induced Coriolis force acting on a vibrating structure. A gyroscope mounted on a moving body, outputs gyroscope signals

$$y_{\omega_t} = \omega_t + e_{\omega_t}, \quad (2.33)$$

where ω_t is the angular velocity of the body w.r.t., an inertial frame and e_{ω_t} is noise. Due to unmodeled effects in the sensors, a bias, b_{ω_t} , could be added

to (2.33) giving

$$y_{\omega_t} = \omega_t + \mathbf{b}_{\omega_t} + \mathbf{e}_{\omega_t}, \quad (2.34)$$

and $\mathbf{b}_{\omega_t}$ could also be included in the state vector with slowly varying dynamics. It is here assumed that the local, flat, navigation frame is inertial since otherwise (2.33) has to include another two components for the earth rotation rate and the turn rate of the navigation frame w.r.t., to earth. This assumption is good only if navigation is not performed over large distances over earth and that the earth rotation rate, $\omega^{ei} = \frac{2\pi}{24}\text{h}^{-1} \approx 7.27 \cdot 10^{-5}\text{rad/s}$, is small compared to the sensor noise. Furthermore, sensor alignment, orthogonality of sensor axes, sensor gain and other parameters are assumed known and accounted for by a factory calibration.

2.3.2 Accelerometers

Accelerometers using MEMS technology are based upon measuring deflections of a cantilever beam to which a so-called proof mass is attached. The mass deflects the beam when external acceleration along the sensitivity axis is applied to the sensor. Accelerometers do not only measure acceleration when used on earth, since they are subject to the gravitational field. Hence, accelerometers measure the specific force which is a sum of the free acceleration and the gravity field. A measurement model is then

$$y_a = \mathbf{a}_t^b - \mathbf{g}^b + \mathbf{e}_{a_t} = \mathbf{R}_t^{be} (\mathbf{a}_t^e - \mathbf{g}^e) + \mathbf{e}_{a_t}, \quad (2.35)$$

where $\mathbf{a}_t^e$ is the acceleration in the navigation frame $\mathbf{g}^e$ is the local gravity vector and $\mathbf{e}_{a_t}$ is measurement noise. Since the local navigation frame is considered to be inertial, Coriolis and centripetal acceleration can be neglected. It is clear that the measured specific force depends on the attitude $\mathbf{R}^{be}$ of the platform and as a consequence, errors in attitude will introduce so-called gravity leakage into the measurements. In case the body frame does not coincide with the sensor a Coriolis term $\boldsymbol{\omega} \times \boldsymbol{\omega} \times \mathbf{r}^{IMUb}$ have to be added to (2.35) as done in Paper C, where $\mathbf{r}^{IMUb}$ is the offset vector, and the result must also be rotated by the relative rotation between these two systems. In the papers considering IMU and monocular vision the body center is in the origin of the IMU thus the offset and rotation between the frames are needed in the camera measurement equation.

2.4 Magnetometers

A simple magnetometer measurements model is

$$y_m = \mathbf{R}_t^{be} \mathbf{m}^e + \mathbf{e}_{m_t}, \quad (2.36)$$

where $\mathbf{m}^e$ is the local earth magnetic field vector and $\mathbf{e}_{m_t}$ is the measurement noise. This model can be used for computing the magnetic north if $\mathbf{m}^e$ is known. In Paper A magnetometers are used slightly differently. Here the objective is to track a magnetic object in order to determine locations of the sensor-nodes containing magnetometers in a sensor network. In this case the object is a vessel

with known magnetic signature and it is modelled as a single dipole $\mathbf{m} = R^{eb} \mathbf{m}^b$ giving rise to the magnetic flux density

$$\mathbf{y}_m = \frac{\mu_0}{4\pi|\mathbf{r}|^5} (3\mathbf{r}(\mathbf{m}^T \mathbf{r}) - |\mathbf{r}|^2 \mathbf{m}), \quad (2.37)$$

where μ_0 is the permeability of the medium and $\mathbf{r}$ is the distance from the sensor to the vessel. The magnetic flux due to the dipole strength decays cubically with the distance. It is here assumed that the effect of the local magnetic field is measured and accounted for before the vessel enters the survey area. For a thorough description on tracking of metallic objects Wahlström (2013) is recommended.

2.5 Vision Sensors

Digital vision sensors capture incident rays of light from the scene onto a sensor array through an optical system. Light reflected by objects in a scene can be mathematically described by *bidirectional reflectance distribution functions* (BRDF). The physical nature of light and image formation is a complex topic which is well beyond the scope of this thesis. There are mainly two manufacturing techniques for digital vision sensors. The first is semiconductor *charge coupled device* (CCD) in which each analog photo sensor (pixel) produces an electrical charge from the incoming light. The second sensor is *complementary metal oxide semiconductor* (CMOS) which directly converts light into voltage. CCD sensors are usually of global shutter type meaning that image is sampled at one time instant. In contrast, CMOS sensors usually sample row by row in a so-called rolling shutter fashion. Rolling shutters are far more complicated to use in geometrical vision problems when either the camera or scene is moving and we will therefore only consider global shutter cameras.

2.5.1 The Pinhole Camera

The by far most commonly used camera model is the pinhole camera. It is a mathematical model describing how points in $\mathbb{R}^3$ relate to points in $\mathbb{R}^2$ on the image plane through a central projection. The model has five intrinsic, or internal, parameters, and for most systems it is an approximation. Additionally, the position and orientation of the camera centre with respect to some other reference frame is also needed if the camera is part of a system, e.g., a stereo rig. Furthermore, most lenses introduce artifacts such as radial and tangential distortion which are important to compensate for.

Figure 2.2 illustrates a frontal pinhole projection. In the pinhole camera model the coordinate frames are:

- c – Camera frame with the *optical centre* as the origin where its z – axis coincides with the *optical axis*, also known as the *principal axis*.
- w – World frame.
- i – Image frame. The *image plane* is orthogonal to the *optical axis* and has

Figure 2.2: Pinhole projection with image the plane placed in front of the camera centre.

the *principal point* h as its origin. The distance between the c and the i frame is called the *focal length* f .

- n – Normalised image frame.
- p – Pixel frame with center in the upper left corner as viewed from the optical centre c .

A point $\mathbf{m}^c = [x^c, y^c, z^c]^T$ in the c -frame relates to a point $\mathbf{m}^n = [x^n, y^n]^T$ in the n -frame as

$$\begin{bmatrix} x^n \\ y^n \end{bmatrix} = \frac{f}{z^c} \begin{bmatrix} x^c \\ y^c \end{bmatrix}. \quad (2.38)$$

It is convenient to write (2.38) as a linear system which can be done by appending unit elements to the vectors giving the homogeneous coordinate representation

$$\begin{bmatrix} x^n \\ y^n \\ 1 \end{bmatrix} \propto z^c \begin{bmatrix} x^n \\ y^n \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}}_{\Pi_0} \begin{bmatrix} x^c \\ y^c \\ z^c \\ 1 \end{bmatrix}. \quad (2.39)$$

With focal length $f = 1$ the normalised image frame n and the image frame i coincides. The projection (2.39) can be expressed as

$$\lambda \mathbf{m}^n = \Pi_0 \mathbf{m}^c, \quad (2.40)$$

where $\lambda \in \mathbb{R}_+$ is an arbitrary scale factor. It is also convenient to define the projection as a function $P([X, Y, Z]^T) = [X/Z, Y/Z]^T$ which is as a map in Euclidean space $P : \mathbb{R}^3 \rightarrow \mathbb{R}^2$. A projection model which is suitable for omnidirectional cameras is the spherical perspective projection which has a scale described by $\lambda = \sqrt{X^2 + Y^2 + Z^2}$, whereas in a planar perspective projection $\lambda = Z$. This means that same equations can be used for both cases and only the depth is

described differently. More advanced models for omnidirectional cameras are accounted for in Scaramuzza et al. (2006)

The digitalisation of the image plane i is the pixel plane p

$$\begin{bmatrix} x^p \\ y^p \end{bmatrix} = \begin{bmatrix} f_x & s_\alpha \\ 0 & f_y \end{bmatrix} \begin{bmatrix} x^i \\ y^i \end{bmatrix} + \begin{bmatrix} h_x x^i \\ h_y y^i \end{bmatrix}, \quad (2.41)$$

or in homogeneous coordinates

$$\begin{bmatrix} x^p \\ y^p \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} f_x & s_\alpha & h_x \\ 0 & f_y & h_y \\ 0 & 0 & 1 \end{bmatrix}}_K \begin{bmatrix} x^i \\ y^i \\ 1 \end{bmatrix}, \quad (2.42)$$

where K is referred to as the *intrinsic parameter matrix* or simply the *calibration matrix*. The parameters of the calibration matrix are the focal lengths in pixel units $f_x = s_x/f$ and $f_y = s_y/f$ where f is the focal length expressed in meters per pixel and s_x, s_y are the pixel sizes in metric units. The center of the image plane $\mathbf{h} = [h_x \ h_y]^T$, is the principal point coordinate which is shifted w.r.t., to the pixel frame which has its origin in the upper left corner, see Figure 2.2. The location of the optical center is not needed for pure computer vision problems but it is important when vision sensor are combined with, for instance, inertial sensors. The skew parameter $s_\alpha = f_x \tan \alpha$ can safely be assumed $s_\alpha \approx 0$ (Hartley and Zisserman, 2004) in most cameras.

Cameras, especially in the lower price-range, suffer from non-negligible distortion due to the optical lenses involved and the mounting of the sensor. This can to some extent be compensated for by tangential-, radial- and pincushion distortion models (Ma et al., 2003). Camera matrices and distortion models can be estimated in standard camera calibration software, see e.g., Bouguet (2010); Zhang (2000).

Figure 2.3: Camera to world transformation.

Points in another frame, say w , can be expressed in the camera frame c . In the pinhole camera model, such transformation is called extrinsic since it does not depend on the intrinsic camera calibration matrix K . Figure 2.3 describes the relation between a point m^w in world coordinates w expressed in camera coordi-

nates c

$$\mathbf{m}^c = \mathbf{R}^{cw}(\mathbf{m}^w - \mathbf{c}^w) = \mathbf{R}^{cw}\mathbf{m}^w - \mathbf{R}^{cw}\mathbf{c}^w, \quad (2.43)$$

which can be written in homogeneous coordinates as

$$\begin{bmatrix} x^c \\ y^c \\ z^c \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R}^{cw} & -\mathbf{R}^{cw}\mathbf{c}^w \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} x^w \\ y^w \\ z^w \\ 1 \end{bmatrix}. \quad (2.44)$$

Combining the extrinsic and intrinsic parameters, coordinates in the world frame can be expressed in pixel coordinates as

$$\begin{bmatrix} x^p \\ y^p \\ 1 \end{bmatrix} \propto z^c \begin{bmatrix} x^p \\ y^p \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & s_\alpha & h_x \\ 0 & f_y & h_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{R}^{cw} & -\mathbf{R}^{cw}\mathbf{c}^w \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} x^w \\ y^w \\ z^w \\ 1 \end{bmatrix}. \quad (2.45)$$

In compact notation (2.45) can be written as

$$\mathbf{x}^p = P\mathbf{x}^w, \quad (2.46)$$

and the matrix P is often just called the camera. In the problems studied the calibration matrix and distortion models are known which allows to working directly on normalised camera measurements $\mathbf{m}^n$ and thus the pinhole camera works as a projective map in Euclidean space.

2.6 Camera Measurement Models

Parametrisation of camera measurements can be done in several ways and the most suitable choice typically depends on the application at hand. For instance, in a filtering context with explicit landmark coordinates included in the state vector measurement models are naturally expressed using the state. Geometrical properties, such as the epipolar constraint, can be used to define implicit measurement models. In this section three different parameterisations that are used in the publications are described.

2.6.1 Direct Parametrisation

From the rigid body transformation in (2.43), the projection of a single measurement of a landmark $\mathbf{m}^w$ onto the normalised image plane is on the form

$$\mathbf{y}^m = P(\mathbf{m}_t^c) + \mathbf{e}_t^m = P(\mathbf{R}_t^{cw}(\mathbf{m}^w - \mathbf{c}_t^w)) + \mathbf{e}_t^m \quad (2.47)$$

where $\mathbf{c}_t^w$ is the position of the moving camera in the world frame and $\mathbf{e}_t^m$ is the so-called re-projection error. This is the preferred error being minimised since it has a sound interpretation and it is straightforward to include intrinsic and extrinsic parameters. We call this the direct parametrisation and it generalises

to n landmark measurements as

$$\begin{bmatrix} y^{m_1} \\ \vdots \\ y^{m_n} \end{bmatrix} = \begin{bmatrix} P(R_t^{cw}(\mathbf{m}_1^w - \mathbf{c}_t^w)) \\ \vdots \\ P(R_t^{cw}(\mathbf{m}_n^w - \mathbf{c}_t^w)) \end{bmatrix} + \mathbf{e}_t^{m_{1:n}}, \quad (2.48)$$

which is the common situation in most practical applications. Depending on which parameters are sought the model can be used in different ways. If, for instance, only landmarks are unknown, then the problem is referred to as mapping or reconstruction and if only the camera pose is unknown the corresponding problem is called localisation or navigation.

2.6.2 Inverse Depth

In (2.47) the measured world point is represented naturally by its Euclidean coordinate. This may however be a poor choice if some points are at a much greater distance from the camera than other points in the scene. This is because the re-projection error cost function becomes very flat in the XYZ -space (Torr and Zisserman, 2000) since the XY coordinates are divided by a large number. This is partly the motivation for using so-called key-frames where only images with sufficient baseline are used for triangulation, see e.g., (Nistér, 2000) for a discussion. The relation to stereo cameras is that range resolution is inversely proportional to the baseline between the cameras and the corresponding change in disparity (image difference for a stereo pair). This phenomena have severe effect in the context of filtering since linearisation errors will be large and may cause the filter to diverge. Bryson et al. (2010) tackles this by tracking landmark coordinates to find the two views with maximal angular separation, parallax, for each landmark and then triangulate the landmark depth based on these two observations alone, known as delayed initialisation. A slightly different approach was proposed in Montiel and Davison (2006) who introduced the Inverse Depth Parametrisation (IDP). It uses six parameters where the first three are the coordinate of the camera from which the landmark was first observed $\mathbf{c}_t^w$ and the three parameters describing the vector from the camera to the landmark encoded by two angles φ^w, θ^w and the inverse depth ρ^w

$$\mathbf{m}^w = \mathbf{c}_t^w + \frac{1}{\rho^w} \mathbf{d}(\varphi^w, \theta^w), \quad (2.49a)$$

$$\mathbf{d}(\varphi^w, \theta^w) = \begin{bmatrix} \cos \varphi^w \sin \theta^w \\ \sin \varphi^w \sin \theta^w \\ \cos \theta^w \end{bmatrix}. \quad (2.49b)$$

The angles are computed from the normalized pixel coordinates as

$$\mathbf{g}^w = \mathbf{R}_t^{wc} \mathbf{m}^n, \quad (2.50)$$

$$\varphi^w = \arctan2(g_y^w, g_x^w), \quad (2.51)$$

$$\theta^w = \arctan2(\| [g_x^w \ g_y^w]^T \|_2, g_z^w), \quad (2.52)$$

where $\arctan2$ is the four-quadrant arctangent function and the inverse depth is initiated with an educated guess. IDP has a small linearisation error even for large uncertainty in depth and it is easy to represent the range of depth uncertainty including infinity in a confidence region. Also, using IDP delayed landmark initialisation is no longer necessary. Obviously more states are needed, however, as soon as depth uncertainty is small enough the IDP landmarks can be converted into standard Euclidean coordinates. The corresponding measurement equation at time t for a landmark initiated at time j in the camera frame is

$$\mathbf{m}_t^c = \mathbf{R}_t^{cw} \left(\rho_t^w \left(\mathbf{c}_t^w - \mathbf{c}_j^w \right) + \mathbf{d}(\varphi_t^w, \theta_t^w) \right), \quad (2.53a)$$

$$y_t^c = P(\mathbf{m}_t^c). \quad (2.53b)$$

The IDP model, including sensor frame offset, was used for fusion with IMU measurements in Paper B. The same sensor setup was used in (Pinies et al., 2007) showing that feature initialisation and prediction in difficult cases, such as forward motion, can be handled better using IDP with support of IMU.

2.6.3 Epipolar Geometry

An important concept in computer vision is how to relate the relative pose of two cameras through point observations. This is called the epipolar constraint and is usually credited to the publication of Longuet-Higgins (1981) and is described in most books on computer vision, see e.g., Ma et al. (2003) or Hartley and Zisserman (2004). Given two images acquired from different vantage points and let $\mathbf{R}$ and $\mathbf{c}$ denote the rotation and translation from camera from the first to the second camera, respectively. Without loss of generality the first camera is located at the origin with no rotation. Thus, a world coordinate expressed in the first camera is simply $\mathbf{m}^w = \mathbf{m}_1^c$. The same point in the second camera is then related by a rigid body transformation

$$\mathbf{m}_2^c = \mathbf{R} \mathbf{m}_1^c + \mathbf{c}. \quad (2.54)$$

Since camera coordinates are related to homogeneous image coordinates through their unknown depths λ as $\mathbf{m}^c = \lambda \mathbf{m}^n$ we have

$$\lambda_2 \mathbf{m}_2^n = \mathbf{R} \lambda_1 \mathbf{m}_1^n + \mathbf{c}. \quad (2.55)$$

The intuition is that the three vectors connecting the two cameras and the point all lie in the same plane and therefore the triple product of the vectors is zero. The triple product is the inner product of one vector with the cross product of the other two. In this case it can be constructed by first eliminating the translation from the right hand side of (2.55) using

$$\mathbf{c} \times \mathbf{c} = 0 \quad (2.56)$$

from left. Then multiply from left with $(\mathbf{m}_2^n)^T$ to construct the two triple products

$$(\mathbf{m}_2^n)^T \mathbf{c} \times \lambda_2 \mathbf{m}_2^n = (\mathbf{m}_2^n)^T \mathbf{c} \times \mathbf{R} \lambda_1 \mathbf{m}_1^n. \quad (2.57)$$

in which the left hand side is zero since $\mathbf{c} \times \lambda_2 \mathbf{m}_2^n$ is perpendicular to $\mathbf{m}_2^n$. Now the epipolar (or bilinear) constraint, is obtained

$$(\mathbf{m}_2^n)^T \mathbf{c} \times \mathbf{R} \lambda_1 \mathbf{m}_1^n = 0, \implies (\mathbf{m}_2^n)^T \mathbf{c} \times \mathbf{R} \mathbf{m}_1^n = 0, \quad (2.58)$$

since λ_1 is non-zero. The product of the translation and the rotation is called the essential matrix

$$\mathbf{E} = \mathbf{c} \times \mathbf{R} = [\mathbf{c}]_{\times} \mathbf{R} \quad (2.59)$$

where $[\mathbf{c}]_{\times}$ is the matrix form of the cross product

$$[\mathbf{c}]_{\times} = \begin{bmatrix} 0 & -c_z & c_y \\ c_z & 0 & -c_x \\ -c_y & c_x & 0 \end{bmatrix}. \quad (2.60)$$

This means that each correspondence gives an equation

$$\mathbf{m}_2^T \mathbf{E} \mathbf{m}_1 = 0, \quad (2.61)$$

where the n superscript has been dropped for notational convenience. Note that $\mathbf{E}$ is homogeneous meaning that it is only defined up to an unknown scale. It has 5 or 6 degrees of freedom, depending on how it is constructed. If the translation is only known up to an unknown positive scale, i.e., it is also a homogeneous quantity, then $\mathbf{E}$ will have 5 DOF. This is typically the case when $\mathbf{E}$ is estimated from image data since the absolute scale of the scene is unknown without other information. If the scaling is known then $\mathbf{E}$ have 6 DOF. In case the calibration of the cameras are unknown the normalised image coordinates will also be unknown and the epipolar constraint then encodes the fundamental matrix $\mathbf{F} = \mathbf{K}_2^{-T} \mathbf{E} \mathbf{K}_1^{-1}$ where $\mathbf{K}_1$ and $\mathbf{K}_2$ denote the calibration of the cameras at the two views. The epipolar constraint is a generic property which may hold for any two vantage points if there are correspondences. It can therefore be used to compactly represent information in terms of a rigid body transformation up to a scale of the translation.

Epipolar geometry can also be used for simplifying correspondence search and verification. If the relative pose is known, a point in the first frame corresponds to a line in the other one, and vice versa, according to

$$\mathbf{l}_2 \sim \mathbf{E} \mathbf{m}_1, \quad (2.62a)$$

$$\mathbf{l}_1 \sim \mathbf{E}^T \mathbf{m}_2, \quad (2.62b)$$

where $\sim$ account for the unknown scale. This also highlights that cameras are bearing sensors. Thus, one may search for the correspondence along a line in the other image, see Figure 2.4. With uncertainty in the relative pose a band is the typical search region. Correspondence candidates may also be deemed correct or false by evaluating a cost defined by the point's distance to the line $\|\mathbf{l}_2^T \mathbf{m}_2\|$ where $\mathbf{l}_2$ is normalised. This is also the basis for efficient correspondence search using stereo cameras.

Figure 2.4: The two top figures show five of the SIFT matches between the two frames which are were used to compute the essential matrix. The bottom left image shows the matches in the first frame and the bottom right figure epipolar lines with matches in the second frame.

Essential Matrix Estimation

Essential matrix estimation is used to find the relative pose between cameras from image correspondences such that the epipolar constraint is (2.61) is satisfied for all correspondences. The linear eight-point algorithm, see e.g., (Ma et al., 2003)[p. 121] is a simple essential matrix estimation algorithm. The algorithm computes an estimate of the essential matrix from a minimum of eight correspondences by computing a singular value decomposition (SVD). This starts from noting that the epipolar constraint can be transformed

$$\mathbf{m}_2^T E \mathbf{m}_1 = 0 \iff (\mathbf{m}_1 \otimes \mathbf{m}_2)^T E^s = 0 \quad (2.63)$$

where $E^s \in \mathbb{R}^{9 \times 1}$ is the stacked matrix of column vectors of E and $\otimes$ denotes the Kronecker product. Using n correspondences results in

$$\underbrace{\begin{bmatrix} (\mathbf{m}_1^1 \otimes \mathbf{m}_2^1)^T \\ \vdots \\ (\mathbf{m}_1^n \otimes \mathbf{m}_2^n)^T \end{bmatrix}}_A E^s = 0 \quad (2.64)$$

which is linear in the unknown E^s . An approximation of the unit length E^s is the solution to (2.64) which is given by minimising $\|AE^s\|$. This is done by computing the SVD of A and define E^s as the singular vector corresponding to the smallest singular value. The required number of correspondences is eight since E has nine entries which are only defined up to scale which means that there are only eight unknowns. Due to noise, the recovered matrix will often not lie in the essential space, however, a solution is to project the matrix onto this space. To further suppress noise the number of correspondences should be more than eight.

There are however situations where the solution to the algorithm is not well defined if e.g., all correspondence points lie in the same plane or the baseline $c \approx 0$. This is partly because the actual degrees of freedom are not taken into account in the algorithm since it considers eight unknowns. Another weakness is the sensitivity to outliers since there is no mechanism for handling these directly. Despite these limitations, the eight-point algorithm was successfully used in Paper D and Paper E for initialisation of rotations without considering potential outliers. However, outliers was later removed in an iterative fashion using IMU data.

The result from eight-point-like methods can be improved by minimising the re-projection error (2.47) using nonlinear optimisation. This requires the landmark coordinates to be known. Instead a suitable error which is only parametrised by the essential matrix is a so-called Sampson error. For n correspondence $\{\mathbf{m}_1^i \leftrightarrow$

$\mathbf{m}_2^i\}_{i=1}^n$ it looks like

$$V(E) = \sum_{i=1}^n \frac{\left((\mathbf{m}_2^i)^T E \mathbf{m}_1^i \right)^2}{(E \mathbf{m}_1^i)_1^2 + (E \mathbf{m}_1^i)_2^2 + (E^T \mathbf{m}_2^i)_1^2 + (E^T \mathbf{m}_2^i)_2^2}, \quad (2.65)$$

where $(E\mathbf{m})_i$ picks out the i -th entry of the epipolar line $E\mathbf{m}$. This Sampson error is a first order approximation of the re-projection error, see e.g., Hartley and Zisserman (2004), and it is extensively used in geometric computer vision problems.

An approach which exploits the 5 DOF imposed by the construction of the essential matrix is the five-point algorithm proposed by (Nistér, 2004). The idea Nister used was that the essential matrix has two non-zero eigenvalues which are equal and this gives rise to a cubical matrix constraint. The algorithm computes the coefficients of a tenth degree polynomial and finds its roots. A perhaps simpler implementation can be found in (Li and Hartley, 2006) which solves the tenth degree polynomial using a hidden variable resultant. An important observation is that the five point method seems less sensitive than the eight-point algorithm to cases when e.g., all points lie on a plane. It also attains higher accuracy because a minimal solver may better exploit the geometric constraints (Li and Hartley, 2006).

RANSAC

Efficient parameterisation which allows fast and/or closed form solutions plays an important role in computer vision algorithms. This is because the set of correspondences between any two images will often contain gross outliers due to association errors which need to be handled. This is usually done by evaluating several candidate models in a RANSAC loop (Fischler and Bolles, 1981). In practice, higher accuracy is obtained by evaluating as many candidates as possible and this is why fast solutions are important since it allows more trials given the computational resources. The acronym RANSAC comes from RANdom Sampling And Consensus Fischler and Bolles (1981) and its paradigm has had a remarkable impact in computer vision. It is an iterative method which aims at finding a model such that the number of inliers is maximised. All that is needed in its basic form is a set of correspondence candidates and some error metric to evaluate the model.

The RANSAC algorithm starts by randomly selecting the minimal set needed to compute a model and then all the other correspondences are evaluated by the error metric. A correspondence pair is labeled as an inlier if it falls below a pre-defined threshold and is otherwise considered to be an outlier. For each model the number of inliers is saved and this is often referred to as scoring. The Sampson error (2.65) is a popular candidate for scoring the essential matrices within the RANSAC loop. The model with the maximum number of inliers, i.e., the largest consensus set, is considered to be the best. Optionally, a final step is to estimate the model again using all the inliers by e.g., minimising (2.65).

The greatest advantage of RANSAC is the ability to robustly estimate the parameters of a model when the measurements are contaminated with outliers while simultaneously finding an inlier set. The number of samples needed for a robust estimate also depends on the number of correspondences needed to estimate the model. It is therefore desirable to have a minimal parametrisation since it is then more likely that samples only contains inliers. It can also be used only as a sampling based method for outlier rejection. A clear limitation is that there is no bound on how many times the loop has to be run in order meet some stopping criterion such as a specific inlier/outlier ratio. RANSAC typically works poorly if there are less than fifty percent of inliers. Despite (or due to) these limitations there are probably hundreds of RANSAC versions published which have different characteristics. For instance, the Maximum Likelihood version called MLESAC (Torr and Zisserman, 2000) is a good option which modifies the score by penalizing outliers.

An example of essential matrix estimation using RANSAC is shown in Figure 2.5 where the integrated yaw angle of the estimated rotation matrices is compared with the yaw angle obtained by integrating the gyroscope signal. The yaw angle estimate from the camera works well for the first 15 seconds and then deteriorates rapidly from the integrated yaw angle of the gyroscope. The gyroscope can be considered a reliable ground truth for this short experiment. This highlight some of the different characteristics of the two sensors. Gyroscopes can handle fast dynamics but may deteriorate over time, due to noise integration, while cameras may have problems with fast dynamics they can be used to estimate orientation accurately for moderate dynamics over long time.

2.7 Computer Vision

The intention with this section is only to provide some basic concepts of computer vision. The interested reader is referred to the two survey papers by Fraundorfer and Scaramuzza (2012); Scaramuzza and Fraundorfer (2011) and references therein, for a modern and detailed exposition.

In order to use the camera measurement models some image processing is needed. Image characteristics having strong response in a detector are used to define so-called interest points. For this to work, the scene has to have some structure. The most commonly used detectors are so-called blob-detectors based on *difference of Gaussians* (DoG) such as the *scale invariant feature transform* (SIFT) (Lowe, 1999) or based on distinctive corners such as the Harris detector (Harris and Stephens, 1988).

The process of tracking interest points over several image frames, also known as correspondence generation, relies on selecting well localised interest points and some local interest point description which can be located in the following frames. Although we have not used patch based methods for tracking in the publications a brief description is given here for completeness. For Harris-like features a patch U around the interest point x can be used to search for its cor-

Figure 2.5: Integrated yaw angle from the 5-point algorithm using RANSAC on the Sampson error compared with the integrated gyroscope measurements on data from a flying platform.

responding location in the next frame J only assuming a simple displacement model in the image

$$J(\mathbf{x} + \mathbf{d}) = U(\mathbf{x}). \quad (2.66)$$

This is a popular model which usually works well for small perspective changes and it is used in most *Kanade-Lucas-Tomasi* (KLT) trackers as explained in Shi and Tomasi (1994). More advanced transformation models for patch matching allowing for rotation, scaling and deformation as

$$J(\mathbf{x} + D\mathbf{x} + \mathbf{d}) = U(\mathbf{x}), \quad (2.67)$$

where $D \in \mathbb{R}^{2 \times 2}$ is a deformation matrix, is particularly useful in longer sequences.

The SIFT detector also stores a feature descriptor for each detected interest point, see Figure 2.6. In the standard setting the descriptor vector has 128 values corresponding to an 8-bin histogram of gradient orientations for each of the 16 local sub-regions around the feature point. The SIFT descriptor can be used for generating correspondences without using any transformation model simply by comparing descriptors using any suitable metric, e.g., the Euclidean distance. This way, matches are found using the appearance of feature descriptors and not their locations. Thus, the set of matches which corresponds to the same feature is called a feature track (Thormählen et al., 2008). Such methods are particularly useful if there is a great deal of uncertainty about motion and/or structure or at loop-closing. In Paper D a purely appearance based approach is used for an initial correspondence search. That is, without assuming anything about the mo-

Figure 2.6: Ten SIFT features are shown with their center location given by the circles. Descriptors computed at different scales with sixteen histograms of gradient orientations indicated by the arrows in each square. Thus, a feature may have multiple orientations depending on their local appearance.

tion between frames correspondences are found only by matching SIFT feature descriptors and not using their point location information. SIFT descriptors and their predicted locations can also be used directly for tracking which was done in Paper B. A key advantage of SIFT is that feature descriptors are rather insensitive w.r.t., changes in scale, illumination, rotation and even viewpoint changes up to 60° (Fraundorfer and Scaramuzza, 2012). These properties make SIFT suitable for matching over wide baselines which also implies that good tracking results and essential matrix estimates can be obtained using a lower sampling frequency compared to KLT-like trackers.

3

Estimation

Estimation is the problem of taking measured data, y_t , to reveal properties, x_t and θ , of systems which are of interest. Basic ingredients are model structures representing the assumptions made on the interaction and time evolution of x_t , θ and y_t . As tools for solving estimation problems, state space models

$$x_t = f(x_{t-1}, u_t, \theta) + w_t, \quad (3.1a)$$

$$y_t = h(x_t, \theta) + e_t, \quad (3.1b)$$

are commonly used model structures. State-space formulations can also be described in terms of their *probability density functions* (PDFs)

$$x_t \sim p(x_t | x_{t-1}, \theta), \quad (3.2a)$$

$$y_t \sim p(y_t | x_t, \theta), \quad (3.2b)$$

where (3.2a) is known as the state transition density, (3.2b) is known as the measurement likelihood and the symbol $\sim$ corresponds to a distribution relation.

3.1 Sensor Fusion

Sensor fusion is the process of combining multi-sensory data in a clever way to obtain a filtered state estimate, $\hat{x}_t$, or a state sequence estimate $\{\hat{x}_i\}_{i=0}^t = \hat{x}_{0:t}$. A state-space model is the key component of many sensor fusion algorithms and model parameters, θ , are usually not of interest. Sensor fusion often use *maximum likelihood* (ML) (Fisher, 1912) estimators which have the form

$$\hat{\theta}^{ML} = \arg \max_{\theta} p(Y|\theta). \quad (3.3)$$

Another important estimator is the *maximum a posteriori* (MAP)

$$\hat{\theta}^{MAP} = \arg \max_{\theta} p(\theta|Y) = \arg \max_{\theta} \frac{p(Y|\theta)p(\theta)}{p(Y)} = \arg \max_{\theta} p(Y|\theta)p(\theta), \quad (3.4)$$

where the second equality is known as Bayes' theorem and the last equality is using the fact that the maximising argument is independent of the normalising constant $p(Y)$.

3.1.1 Smoothing and Filtering

The smoothed state $\hat{x}_{0:t}$ defines an estimate of the whole state trajectory and it can be obtained as the MAP estimate of the state sequence given the measurements

$$\hat{x}_{0:t} = \arg \max_{x_{0:t}} p(y_{0:t}|x_{0:t})p(x_{0:t}). \quad (3.5)$$

With Gaussian noise and initial state $x_0 \sim \mathcal{N}(\bar{x}_0, P)$ the negative log of (3.5) times two becomes a Gaussian MAP estimation problem

$$\begin{aligned} \hat{x}_{0:t} = \arg \min_{x_{0:t}} \quad & \|\bar{x}_0 - x_0\|_{P^{-1}}^2 + \sum_{i=1}^t \|x_i - f(x_{i-1})\|_{Q^{-1}}^2 + \sum_{i=1}^t \|y_i - h(x_i)\|_{R^{-1}}^2 = \\ & \arg \min_{x_{0:t}} \quad V(x_{0:t}), \end{aligned} \quad (3.6)$$

where the terms not directly depending on $x_{0:t}$ have been left out, see for instance Rao (2000) for details. This is also a nonlinear least squares formulation, a topic which will be treated in Section 4.3.3. Note that if only a part of the whole batch is considered at each time step a so-called moving horizon estimation problem is obtained. Alternatively, the process and measurements can be viewed as equality constraints and the constrained formulation of (3.6) is then

$$\hat{x}_{0:t} = \arg \min_{x_{0:t}} \quad \|\bar{x}_0 - x_0\|_{P^{-1}}^2 + \sum_{i=1}^t \|w_i\|_{Q^{-1}}^2 + \sum_{i=1}^t \|e_i\|_{R^{-1}}^2, \quad (3.7a)$$

$$\text{subject to} \quad x_i = f(x_{i-1}) + w_i, \quad (3.7b)$$

$$y_i = h(x_i) + e_i. \quad (3.7c)$$

MAP estimation can be extended to include other parameters in f and h , besides the state, resulting in a joint smoothing and parameter estimation problem. The particular benefit with this optimisation viewpoint is that it is straightforward to add constraints which is not easily done in a filtering context.

In the case of linear dynamics and linear measurement equations (3.6) becomes a convex optimisation problem that can be efficiently implemented as a *Rauch-Tung-Striebel* (RTS) smoother Rauch et al. (1965). A straightforward, yet approximate, extension to nonlinear systems is given by the *extended-RTS* (E-RTS) smoother where the forward filter is realised using an EKF. This method is used in Paper E. Note that in this nonlinear setting the E-RTS may be improved using iterations, step control, and other techniques, since in the end we are just solving a nonlinear optimisation problem.

Algorithm 1 Extended Kalman Filter

Available measurements are $Y = \{y_1, \dots, y_N\}$. Require an initial state, $\hat{x}_{0|0}$, and an initial state covariance, $P_{0|0}$, and use the models (3.1).

1. Time Update

$$\hat{x}_{t|t-1} = f(\hat{x}_{t-1|t-1}, u_t), \quad (3.8a)$$

$$P_{t|t-1} = F_t P_{t-1|t-1} F_t^T + Q_t, \quad (3.8b)$$

2. Measurement Update

$$S_t = H_t P_{t|t-1} H_t^T + R_t, \quad (3.9a)$$

$$K_t = P_{t|t-1} H_t^T S_t^{-1}, \quad (3.9b)$$

$$\hat{x}_{t|t} = \hat{x}_{t|t-1} + K_t (y_t - h(\hat{x}_{t|t-1})), \quad (3.9c)$$

$$P_{t|t} = P_{t|t-1} - K_t H_t P_{t|t-1}. \quad (3.9d)$$

Where

$$F_t \triangleq \left. \frac{\partial f(x_t, u_t)}{\partial x_t} \right|_{(x_t, u_t) = (\hat{x}_{t-1|t-1}, u_t)}, \quad H_t \triangleq \left. \frac{\partial h(x_t)}{\partial x_t} \right|_{(x_t) = (\hat{x}_{t|t-1})}, \quad (3.10)$$

while, Q_t and R_t are the covariance matrices of w_t and e_t , respectively.

Extended Kalman Filter

A popular estimator is the extended Kalman filter which is described in many books, see e.g., Kailath et al. (2000). The EKF works in a two step procedure summarised in Algorithm 1. As with the E-RTS there are no convergence guarantees since the involved functions are nonlinear.

3.2 Optimisation

Many batch and filtering problems can be formulated in terms of optimisation programs to which there are many software packages readily available. A quite general optimisation program is

$$\underset{\theta}{\text{minimise}} \quad V(\theta) \quad (3.11a)$$

$$\text{subject to} \quad c_E(\theta) = 0 \quad (3.11b)$$

$$c_I(\theta) \leq 0, \quad (3.11c)$$

where $V : \mathbb{R}^n \rightarrow \mathbb{R}$ is the objective function, c_E are equality constraints, c_I are inequality constraints and θ are the variables.

The Lagrange function, often just called the Lagrangian, is obtained by taking the optimisation problem (3.11) and augmenting the objective function with a weighted sum of the constraints as

$$L(\theta, \lambda, \nu) = V(\theta) + \lambda^T c_E(\theta) + \nu^T c_I(\theta), \quad (3.12)$$

with associated dual variable vectors, λ and ν . The first order necessary optimal-

ity conditions are

$$\nabla V(\theta) + \lambda^T \nabla c_E(\theta) + \nu^T \nabla c_I(\theta) = 0 \quad (3.13a)$$

$$c_E(\theta) = 0 \quad (3.13b)$$

$$c_I(\theta) \leq 0 \quad (3.13c)$$

$$\nu \geq 0 \quad (3.13d)$$

$$\text{diag}(\nu)c_I(\theta) = 0 \quad (3.13e)$$

which have to be satisfied at the optimum $(\theta^*, \lambda^*, \nu^*)$ and (3.13a) is the gradient of the Lagrangian (3.12). These equations are often referred to as the *Karush-Kuhn-Tucker* (KKT) conditions, see e.g., (Boyd and Vandenberghe, 2004; Nocedal and Wright, 2006) for detailed explanations and some historical notes. For convex problems the KKT conditions are also sufficient whereas for non-convex problems a KKT point is merely a candidate solution.

Specific classes of problems can be identified depending on e.g., the properties of V , the choice of variables, among others. If, for example, the objective function is $V(\theta) = \frac{1}{2}\theta^T G\theta + \theta^T c$, G is a symmetric matrix, and the constraints are linear in θ , then a *quadratic program* (QP) is obtained. Many problems can be transformed into an equivalent QP and an instructive example on linear Kalman filtering is given below.

Example 3.1: KF measurement update as a Quadratic Program

The Kalman filter iterates two equations for the state

$$\hat{x}_{t|t-1} = A\hat{x}_{t-1|t-1} \quad (3.14a)$$

$$\hat{x}_{t|t} = \hat{x}_{t|t-1} + P_{t|t-1}C^T(CP_{t|t-1}C^T + R)^{-1}(y_t - C\hat{x}_{t|t-1}), \quad (3.14b)$$

where (3.14a) is the time update and (3.14b) is the measurement update. Equivalently, the left hand side of the measurement update can be specified as the following QP

$$\{\hat{x}_{t|t}, \hat{e}_t\} = \arg \min_{x, e_t} e_t^T R^{-1} e_t + (x - \hat{x}_{t|t-1})^T P_{t|t-1}^{-1} (x - \hat{x}_{t|t-1}) \quad (3.15a)$$

$$\text{subject to } y_t = Cx + e_t, \quad (3.15b)$$

or alternatively, the unconstrained version is

$$\hat{x}_{t|t} = \arg \min_x (y_t - Cx)^T R^{-1} (y_t - Cx) + (x - \hat{x}_{t|t-1})^T P_{t|t-1}^{-1} (x - \hat{x}_{t|t-1}). \quad (3.16)$$

The actual gain from these formulations, compared to the standard KF equations (3.14), is that constraints can be added easily and modification of the objective function becomes straightforward. Such an example is modelling of so called heavy-tailed noise which is a direct approach to handle non-Gaussian residuals. This can be treated by adding another variable z_t to (3.15), as in (Mattingley and

Boyd, 2010), resulting in

$$\{\hat{x}_{t|t}, \hat{e}_t, \hat{z}_t\} = \arg \min_{x, e_t, z_t} e_t^T R^{-1} e_t + (x - \hat{x}_{t|t-1})^T P_{t|t-1}^{-1} (x - \hat{x}_{t|t-1}) + \lambda \|z_t\|_1 \quad (3.17a)$$

$$\text{subject to } y_t = Cx + e_t + z_t, \quad (3.17b)$$

where $\lambda \geq 0$ is a design parameter controlling how much the $\|z_t\|_1$ -norm (ℓ_1) should be favoured. This is not a QP anymore, however it can be transformed into another QP which has the form

$$\{\hat{x}_{t|t}, \hat{e}_t, \hat{z}_t\} = \arg \min_{x, e_t, z_t} e_t^T R^{-1} e_t + (x - \hat{x}_{t|t-1})^T P_{t|t-1}^{-1} (x - \hat{x}_{t|t-1}) + \lambda \mathbf{1}^T u \quad (3.18a)$$

$$\text{subject to } y_t = Cx + e_t + z_t, \quad (3.18b)$$

$$-u \leq z_t \leq u. \quad (3.18c)$$

For small λ some elements in z_t will be exactly zero, whereas for large λ all elements will be zero and the original problem is obtained. This means that the filter is more robust with respect to non-Gaussian errors. The ℓ_1 -norm can be interpreted in a statistical sense as z_t being Laplace distributed $p(z_t) = \mathcal{L}(0, 2\sigma^2/\lambda)$, see e.g., Hastie et al. (2009). ℓ_1 -regularisation is also used to find sparse parameter estimates and these methods are known as *least absolute shrinking and selection operator* (LASSO) Tibshirani (1996).

Constrained programs can be transformed into unconstrained counterparts using the Lagrangian and additional penalisation terms to account for dropped constraints. Many software packages can solve constrained problems by parsing them as unconstrained ones, or vice versa, if it is more efficient to solve them this way. Popular unconstrained methods are steepest descent, (quasi-) Newton and trust-region, see e.g., Nocedal and Wright (2006); Dennis and Schnabel (1983).

3.2.1 Iterated Extended Kalman Filter

An interesting parallel between filtering and optimisation is that the measurement update in the EKF (3.9) is the solution to the following NLS problem

$$\hat{x}_{t|t} = \arg \min_x \frac{1}{2} \|y_t - h(x)\|_{R_t^{-1}}^2 + \frac{1}{2} \|x - \hat{x}_{t|t-1}\|_{P_{t|t-1}^{-1}}^2 = \arg \min_x V(x) \quad (3.19)$$

which is the nonlinear counterpart of (3.16). The EKF solution is given by a full, single step, in a Gauss-Newton procedure (Bertsekas, 1994). Note that in the filtering case it does not matter if the state dynamics are nonlinear or not. The pure EKF may be a poor choice if e.g., the predicted state is far from the true one. Albeit Gauss-Newton does not promise global (or even local) convergence, iterations may improve the estimate. This approach is used in the *iterated-EKF* (IEKF), see e.g., Jazwinski (1970); Bell and Cathey (1993); Bar-Shalom and Li (1993), in

which the measurement update for the state is iterated a few times

$$H_i = \frac{\partial h(x)}{\partial x} \Big|_{x=x_i}, \quad (3.20a)$$

$$K_i = P_{t|t-1} H_i^T (H_i P_{t|t-1} H_i^T + R_t)^{-1}, \quad (3.20b)$$

$$x_{i+1} = \hat{x}_{t|t-1} + K_i (y_t - h(x_i) - H_i(\hat{x}_{t|t-1} - x_i)), \quad (3.20c)$$

where the first iteration is exactly the same as in the EKF measurement update. The measurement updated state and the covariance approximation is

$$\hat{x}_{t|t} = x_{i+1}, \quad (3.21a)$$

$$P_{t|t} = P_{t|t-1} - K_i H_i P_{t|t-1}, \quad (3.21b)$$

where the updated covariance is only updated using the last Kalman gain and measurement Jacobian, K_i and H_i , respectively. The IEKF assumes that each step reduces the cost (3.19)

$$V(x_{i+1}) \leq V(x_i), \quad (3.22)$$

without evaluating if that was the case. Modifying the update (3.20c) as

$$\begin{aligned} x_{i+1} &= \hat{x}_{t|t-1} + K_i (y_t - h(x_i) - H_i(\hat{x}_{t|t-1} - x_i)), \\ &= x_i + (\hat{x}_{t|t-1} - x_i + K_i (y_t - h(x_i) - H_i(\hat{x}_{t|t-1} - x_i))), \end{aligned} \quad (3.23)$$

a step control parameter α can be introduced to ensure cost reduction. The modified measurement update of the IEKF is then on the form

$$x_{i+1} = x_i + \alpha_i (\hat{x}_{t|t-1} - x_i + K_i (y_t - h(x_i) - H_i(\hat{x}_{t|t-1} - x_i))), \quad (3.24)$$

$$= x_i + \alpha_i p_i, \quad (3.25)$$

where the step length $0 < \alpha \leq 1$ and the search direction p is chosen such that (3.22) is satisfied in each step. We call this approach IEKF-L. More advanced line search strategies that employs conditions on the curvature for sufficient decrease could of course be used. The IEKF with and without line search is illustrated in Example 3.2.

Example 3.2: Bearings Only Tracking and IEKF

An applied example of IEKF bearings-only tracking is studied, which also can be found in Gustafsson (2012) as Example 8.1. For simplicity, the target is stationary, i.e., $w_t = 0$ and $Q = 0$, at the true position $x^* = [1.5, 1.5]^T$. The bearing measurement function from the j -th sensor S^j at time t is

$$y_t^j = h_j(x_t) + e_t = \arctan2(y_t - S_y^j, x_t - S_x^j) + e_t, \quad (3.26)$$

where $\arctan2()$ is the two argument arctangent function, S_y and S_x denotes the y and the x coordinates of the sensors, respectively. The noise is $e_t \sim \mathcal{N}(0, R_t)$.

The Jacobians of the dynamics and measurements are

$$F = I, \quad H^j = \frac{1}{(x - S_x^j)^2 + (y - S_y^j)^2} \begin{bmatrix} -(y - S_y^j) \\ x - S_x^j \end{bmatrix}. \quad (3.27)$$

With the two sensors having positions $S^1 = [0, 0]^T$ and $S^2 = [1.5, 0]^T$. The filter is initialised with $\hat{x}_{0|0} = [1, 1]^T$ and $P_{0|0} = I_2$. In Figure 3.1 the IEKF is compared with the EKF for a single realisation with $e_t \sim \mathcal{N}(0, \pi 10^{-3} I_2)$ for the left plot and ten noise-free measurements for the right plot with the same covariance as in the left plot. If this artificial measurement covariance is decreased by a few orders of magnitude, then the EKF covariance does not capture the true uncertainty but the IEKF does. For this simple example 10 iterations are used since then the IEKF

Figure 3.1: Left: The IEKF is iterated 10 times and comes closer to the true state with a covariance that captures the uncertainty. Right: 10 noise-free measurements are given and the IEKF, again iterated 10 times, converges to the true position much faster than the EKF.

have converged and the result clearly improves the estimate. However, it may be sufficient to perform fewer iterations to speed up execution.

The cost function for this example is nearly convex close to the true target. This means that if we start close enough it should be safe to take apply the IEKF without any modifications. It was however also verified that the actual cost decreased in each step. In the left plot in Figure 3.2 the target is in the same location as before but the initial guess is $\hat{x}_{0|0} = [0.2, 1.5]^T$. Given one measurement from each sensor the EKF performs one update and the IEKF performs 3 iterations. It is obvious that both the target location and the covariance estimate is much better when using the IEKF.

Starting with the initial guess $\hat{x}_{0|0} = [0.3, 0.1]^T$ with $P_{0|0} = 2I_2$ where the curvature of the cost function is bit more difficult. The EKF and IEKF-L are given perfect measurement but with assumed covariances $Q = R = \pi 10^{-3} I_2$. The position estimate is shown in the right plot in Figure 3.2. Note that the IEKF-L covariance is consistent as opposed to the EKF. The rapid convergence of the IEKF-L is not

Figure 3.2: Left: The target estimate and the covariance is much better using the IEKF. The dashed lines corresponds to the iterated estimate in the IEKF. Note that the first step in the IEKF is the same as the EKF. The level curves illustrates the NLS cost with a minimum at the true target position. Right: The EKF has poor performance and under-estimates the covariance while IEKF-L gives good results by step size reduction and just 3 iterations.

surprising since the variable step is based on cost decrease at each iteration. More iterations does not give any significant improvement and the steps gets short.

3.2.2 Nonlinear Least Squares

Nonlinear least squares problems are obtained when the objective function is a sum of squared errors as in the MAP smoothing problem (3.6) or the constrained form (3.7a). Popular solvers are Gauss-Newton and Levenberg-Marquardt which approximates nonlinear functions by appropriate linear functions. The goal in NLS is to estimate parameters, θ , by minimising a parametrised residual vector, $\tilde{\varepsilon}(\theta) \sim \mathcal{N}(0, \Sigma_\theta)$. For notational convenience the residuals are normalised according to their assumed covariance

$$\varepsilon(\theta) = \Sigma_\theta^{-T/2} \tilde{\varepsilon}(\theta), \quad (3.28)$$

where $\Sigma_\theta^{-T/2}$ denotes the transposed matrix square-root and then the Mahalanobis notation can be dropped since $\|\tilde{\varepsilon}(\theta)\|_{\Sigma_\theta^{-1}}^2 = (\Sigma_\theta^{-T/2} \tilde{\varepsilon}(\theta))^T (\tilde{\varepsilon}(\theta) \Sigma_\theta^{-T/2}) = \|\varepsilon(\theta)\|_2^2$. The residuals are said to be minimised in a least-squares sense by the cost

$$V(\theta) = \frac{1}{2} \sum_{t=1}^N \|\varepsilon_t(\theta)\|_2^2 = \frac{1}{2} \sum_{t=1}^N \varepsilon_t^T(\theta) \varepsilon_t(\theta). \quad (3.29)$$

Algorithm 2 Gauss-Newton

-
1. Require initial an estimate θ and Jacobian $J(\theta)$
 2. Compute a search direction p by solving

$$J(\theta)J(\theta)^T p = -J(\theta)\varepsilon(\theta). \quad (3.33)$$
 3. Compute a step length, α , such that the cost (3.29) is decreased. This can be done using line search, see e.g., Nocedal and Wright (2006, page 297) or Boyd and Vandenberghe (2004, page 464).
 4. Update the parameters

$$\theta := \theta + \alpha p. \quad (3.34)$$
 5. Terminate if a stopping criteria is met. Such criteria can be; the change in cost is small, the number of iterations has exceeded some threshold, among others.
 6. Otherwise, return to step 2.
-

Define $\varepsilon(\theta) = [\varepsilon_1(\theta) \ \varepsilon_2(\theta) \ \dots \ \varepsilon_N(\theta)]^T$ and the Jacobian $J(\theta) = \frac{d\varepsilon^T(\theta)}{d\theta}$, then the gradient and the Hessian of (3.29) with respect to the parameters are given by

$$\nabla V(\theta) = \frac{dV(\theta)}{d\theta} = \frac{1}{2} \sum_{t=1}^N \varepsilon_t(\theta) \frac{d\varepsilon_t(\theta)}{d\theta} = J(\theta)\varepsilon(\theta), \quad (3.30)$$

and

$$\frac{d^2 V(\theta)}{d\theta^2} = J(\theta)J(\theta)^T + \frac{1}{2} \sum_{t=1}^N \varepsilon_t(\theta) \frac{d^2 \varepsilon_t^T(\theta)}{d\theta^2}, \quad (3.31)$$

respectively. The extension to multivariable residuals is easily obtained by stacking the vectorisation of the individual residuals which again gives a scalar cost function.

The Gauss-Newton method can be seen as a modified Newton method which applies only to NLS problems. It is computationally cheap since the Hessian of the objective function is approximated as

$$\frac{d^2 V(\theta)}{d\theta^2} \approx J(\theta)J(\theta)^T, \quad (3.32)$$

thus there is no need for second order derivatives. The approximation is good when the initial θ is close to the optimum but it may be bad if some residuals are large. An option is then to include the second order terms in (3.31) or approximate them with some secant method. The Gauss-Newton method as an algorithm is summarised in Algorithm 2. The Gauss-Newton method may encounter problem if the Jacobian is singular or ill-conditioned. A straightforward remedy is given by the Levenberg-Marquardt algorithm which modifies the normal equations as

$$(JJ^T + \mu I)p = -J\varepsilon. \quad (3.35)$$

where μ is a positive number and the θ dependence have been omitted. The μ parameter acts as an interpolation of the Gauss-Newton method and gradient

descent. This can be seen by noticing that Gauss-Newton corresponds to $\mu = 0$ and for large μ the step will be approximately in the gradient descent direction which typically is the preferred option if the initial solution is far from the optimum. The parameter is adaptively updated by some method and should typically decrease as the minimum is approached. Another modification suggested by Marquardt (1963) is to take larger steps in the direction in which the gradient is small by

$$(JJ^T + \mu \text{diag}\{JJ^T\})p = -J\varepsilon, \quad (3.36)$$

and thus speed up convergence.

Note that in practice parameters are often subject to constraints, for instance; unit quaternions should have unit length; physical landmarks need to be in front of the camera (Hartley, 1998); motion is constrained by motion models. When parameters are updated by a simple increment such constraints may be violated. It is therefore important to have a stable local parametrisation for the update followed by some procedure such that the parameters do not violate the constraints, for instance normalisation of an updated quaternion. Such an approach is used in Paper D since the gyroscope rates are natural parameters for the local update and the unit quaternion for global rotation parametrisation.

3.3 Problem Structure

For problems with many parameters the key to efficiency is to utilise the specific structure of each problem in the corresponding Jacobian and the Hessian approximation. Although the normal equations can be computed efficiently by numerical matrix factorisations, such as QR, LDL and SVD it may not be a good option if the factorisation needs to be updated. For large problems even the computation of the normal equations may be infeasible due to e.g., memory or time constraints. Explicit, rather than numerical, factorisations are then a good option. A well-known trick for efficient equation system solving is to use the Schur complement and an example of this is given below.

Example 3.3: Schur Complement

In batch problems such as NLS-SLAM and *bundle adjustment* (BA) there is a natural sparsity in the Jacobian and thus in the Hessian approximation. Partition the normal equations (3.33) as

$$\begin{bmatrix} J_c J_c^T & J_c J_m^T \\ J_m J_c^T & J_m J_m^T \end{bmatrix} \begin{bmatrix} p_c \\ p_m \end{bmatrix} = - \begin{bmatrix} J_c \varepsilon_c \\ J_m \varepsilon_m \end{bmatrix} \iff \begin{bmatrix} H_{cc} & H_{cm} \\ H_{mc} & H_{mm} \end{bmatrix} \begin{bmatrix} p_c \\ p_m \end{bmatrix} = - \begin{bmatrix} J_c \varepsilon_c \\ J_m \varepsilon_m \end{bmatrix} \quad (3.37)$$

where J_c and J_m are the Jacobian w.r.t., the camera poses and landmarks respectively. The primary sparsity structure in BA comes from the fact that each landmark observation only depends on one camera pose which means that H_{cc} and H_{mm} are block diagonal. Solving (3.37) for p_c and p_m can be done efficiently using

the Schur complement of e.g., H_{mm} as

$$(H_{cc} - H_{cm}H_{mm}^{-1}H_{mc})p_c = -J_c\varepsilon_c + H_{cm}H_{mm}^{-1}J_m\varepsilon_m \quad (3.38a)$$

$$p_m = H_{mm}^{-1}(-J_m\varepsilon_m - H_{mc}p_c) \quad (3.38b)$$

where the much smaller reduced system (3.38a) is solved first and p_c is back-substituted into (3.38b). And since H_{mm} is block diagonal, its inverse is cheap to compute. Furthermore, it is often the case that all landmarks are not observed the whole time and this gives rise to a secondary sparsity structure in the H_{cm} matrix which can be exploited.

For very large problems the normal equations may be solved by quasi-Newton methods such as L-BFGS (Nocedal, 1980) which efficiently approximate the inverse Hessian. Other approaches are conjugate gradients or in a distributed way using the *alternating direction method of multipliers* (ADMM) (Boyd et al., 2011). These methods converge slowly, yet they may be the only feasible option for large systems where direct methods are not suitable. In general there are few, if any, guarantees that the global optimum will be found by any nonlinear program. However, careful selection of the initial starting point greatly improves the chance of reaching the optimum.

4

SLAM

In this chapter a brief overview of some SLAM estimation methods is given. More thorough descriptions are given in the three appended SLAM publications.

4.1 Introduction

SLAM problems are combinations of localisation and mapping type problems solved simultaneously. There are mainly two strategies and these are either based on filtering or on batch optimisation. The early research almost exclusively focused on filtering methods which recursively incorporate the measurements estimating the posterior filtering density $p(x_t, m | y_{1:t})$ and the list of filtering based acronyms is long. In summary, many of them are either using particle filters Montemerlo et al. (2002) or extended Kalman filters, Smith et al. (1990). Figure 4.1 illustrates a SLAM setup with a moving platform and observing landmark features from different locations.

4.1.1 Probabilistic Models

The target in SLAM is to either maximise the posterior density of the complete trajectory and map

$$p(x_{0:t}, m | y_{0:t}), \quad (4.1)$$

or the filtering density

$$p(x_t, m | y_{0:t}), \quad (4.2)$$

which is obtained by marginalising old states. Here $x_{0:t}$ is the whole state trajectory, x_t is the current state, m is a static map and are the $y_{0:t}$ measurements relating to the state and the map. For notational convenience, correspondence

Figure 4.1: A moving platform with body coordinate system (b) is observing an environment represented by point landmarks, $m^1, \dots, m^8$. Also, a global, fixed navigation coordinate system, (n), is drawn.

variables and input signals are not made explicit. The SLAM posterior (4.1) can for instance be factorised as

$$p(x_{0:t}, m | y_{0:t}) = p(x_{0:t} | y_{0:t}) p(y_{0:t} | x_{0:t}, m) = p(x_{0:t}) p(y_{0:t}^x | x_{0:t}) p(y_{0:t}^{x,m} | x_{0:t}, m), \quad (4.3)$$

where the first factor is the process model, the second factor is the measurement likelihood independent of the map, e.g., GPS or IMU measurements, and the third is the measurement likelihood of measurements that depend on the map and the process. From (4.3) the filtering and smoothing forms are straightforwardly obtained. As was shown in Section 3.1.1 the smoothing density becomes an NLS problem if the noise sources are Gaussian as is the common assumption in most algorithms. The batch formulation is the target in GraphSLAM (Thrun and Montemerlo, 2006; Thrun et al., 2005). Similar to FastSLAM (Montemerlo et al., 2002, 2003), the posterior is factorised as

$$p(x_{0:t}, m | y_{0:t}) = p(x_{0:t} | y_{0:t}) p(m | x_{0:t}, y_{0:t}), \quad (4.4)$$

where $p(x_{0:t} | y_{0:t})$ is the posterior of trajectories. This density is obtained by marginalising the landmark parameters which introduces links, relative pose constraints, between any two poses measuring the same landmark and the result is a pose graph. This is in close relation to the reduced system in (3.38a) obtained using the Schur complement. GraphSLAM maintains a pose graph as a Gaussian with mean and covariance but only computes the conditional mean of the map using the factorisation

$$p(m | x_{0:t}, y_{0:t}) = \prod_{i=1}^{N_m} p(m^i | x_{0:t}, y_{0:t}), \quad (4.5)$$

which is the same approach as in FastSLAM. This means that each landmark can be treated independently given the true trajectory, thus avoiding to keep track of the full correlation structure of the map which is the main draw-back with EKF-SLAM.

4.2 EKF-SLAM

EKF-SLAM is probably the most common SLAM method and it is often straightforward to implement as described in e.g., Durrant-Whyte and Bailey (2006); Smith et al. (1990). For a thorough treatment, the book by Thrun et al. (2005) serves as a standard reference. In feature based SLAM, coordinates in the global frame are explicitly represented as landmarks, m , which are part of the state vector. The standard assumption is that the landmarks are stationary but dynamic objects can naturally be included in the state vector (Bibby and Reid, 2007). Assume that measurements arrive in the same rate as the dynamic model. Then the landmark and the measurement models are given by

$$x_t = f(x_{t-1}) + w_t, \quad (4.6a)$$

$$m_t = m_{t-1}, \quad (4.6b)$$

$$y_t = h(x_t, m_t) + e_t. \quad (4.6c)$$

The EKF given in Algorithm 1 applies to (4.6) with just a few modifications. The prediction step in EKF-SLAM is given by

$$\hat{x}_{t|t-1} = f(\hat{x}_{t-1|t-1}), \quad (4.7a)$$

$$\hat{m}_{t|t-1} = \hat{m}_{t-1|t-1}, \quad (4.7b)$$

$$P_{t|t-1}^{xx} = F_t P_{t-1|t-1}^{xx} F_t^T + Q, \quad (4.7c)$$

$$P_{t|t-1}^{xm} = F_t P_{t-1|t-1}^{xm}, \quad (4.7d)$$

$$P_{t|t-1}^{mx} = P_{t-1|t-1}^{mx} F_t^T, \quad (4.7e)$$

where

$$F_t \triangleq \left. \frac{\partial f(x_{t-1})}{\partial x_{t-1}} \right|_{(x_{t-1})=(\hat{x}_{t-1|t-1})}. \quad (4.7f)$$

Note that only the vehicle state covariance and the cross terms are updated while that the map mean and covariance remains unchanged. The full covariance matrix is

$$P = \begin{bmatrix} P^{xx} & P^{xm} \\ P^{mx} & P^{mm} \end{bmatrix}. \quad (4.8)$$

When new landmarks are initialised they are appended to the state vector but the vehicle state dimension stays the same. If the map estimation is only used locally for vehicle state estimation i.e., odometry, then old landmarks can be removed

from the filter. The measurement update for EKF-SLAM is given by

$$K_t = P_{t|t-1} H_t^T (H_t P_{t|t-1} H_t^T + R_t)^{-1}, \quad (4.9a)$$

$$\begin{bmatrix} \hat{x}_{t|t} \\ \hat{m}_{t|t} \end{bmatrix} = \begin{bmatrix} \hat{x}_{t|t-1} \\ \hat{m}_{t|t-1} \end{bmatrix} + K_t (y_t - h(\hat{x}_{t|t-1}, \hat{m}_{t|t-1})), \quad (4.9b)$$

$$P_{t|t} = P_{t|t-1} - K_t H_t P_{t|t-1}, \quad (4.9c)$$

where

$$H_t \triangleq \begin{bmatrix} \frac{\partial}{\partial x_t} h(x_t, m_t) & \frac{\partial}{\partial m_t} h(x_t, m_t) \end{bmatrix} \bigg|_{(x_t, m_t) = (\hat{x}_{t|t-1}, \hat{m}_{t|t-1})}. \quad (4.10)$$

The measurement Jacobian (4.10) is often rather sparse since the sensor will typically only observe a part of the landmark state at each time instant and an efficient implementation exploits this structure. Since the measurements are assumed independent the measurement update can be processed iteratively avoiding the need for inverting a large matrix in the Kalman gain computation (4.9a).

In Paper B EKF-SLAM is used for intialisation of the trajectory and map which makes it limited to small problems. In Figure 4.2 the horizontal speed estimate from EKF-SLAM and NLS-SLAM is shown.

Figure 4.2: The smoothed horizontal speed of the camera in red and EKF in blue. The true speed is 0.1 m/s except for when the robot stops and changes direction, this happens at about 4 seconds and 6 seconds.

The EKF-SLAM framework also applies to sensor network calibration, as in Paper A. Obviously there is then no need for correspondence search since the sensor identities (landmarks) are known and the map size is also fixed. As was pointed out in Section 2.4 measurements are here obtained in the sensor node magnetometers assuming a known magnetic dipole model of the survey vessel. This is the reverse measurement relation to the ordinary SLAM concept.

4.3 Batch SLAM

Batch methods, also known as Full SLAM, have recently come to dominate SLAM research both offline and as a sub-system in online applications. Some beneficial properties of batch methods are:

- Loop closing and thus drift compensation is easier.
- The effect of linearisation errors can be reduced through iterative refinement.
- Efficient optimisation routines can be utilised.
- Data association decisions can easily be undone.
- The inherent primary sparsity is utilised.
- The complete map correlation structure does not have to (but can be) computed.

In both filtering and batch SLAM applications the system unavoidably grows with the exploited space since more memory is needed to store the map and possibly historical motion estimates. However, in most realistic scenarios only a few parameters are affected by the measurements and therefore only small parts of the system, or its factorisation, need to be updated. A concrete example is the square-root *smoothing and mapping* (SAM) Dellaert and Kaess (2006) which solves the Full SLAM problem incrementally in real-time.

Efficient solutions to batch formulations are utilising the inherent sparsity in the Jacobian J or equivalently the associated information (inverse covariance) matrix $I = JJ^T$. In contrast, the filter covariance, and likewise the filter information matrix, is full since past vehicle states are marginalised as shown in Paskin (2003). The batch sparsity ideas are exploited in a SLAM context by (M. Kaess and A. Ranganathan and F. Dellaert, 2008; Grisetti et al., 2011; Thrun and Montemerlo, 2006; U. Frese, 2005; Paskin, 2003) and many others. Furthermore, Dellaert and Kaess make no special distinction between SAM and BA since they are both often treated as NLS problems in batch form.

4.3.1 Graphs

Recently, optimisation based batch methods on graphs, and especially pose graphs (Lu and Milios, 1997), have attracted much interest and will therefore be given some attention here as well. The GraphSLAM algorithm, proposed by (Thrun

Figure 4.3: Left: Structure of a Jacobian from the NLS-SLAM problem in Paper B. The upper left diagonal block corresponds to the process derivatives, the left lower block corresponds to the derivatives of the camera measurements w.r.t., the IMU/camera states and the lower right block are derivatives of the camera measurements w.r.t., the landmark parameters. Right: Jacobian matrix from NLS-SLAM problem in Paper D. The structure is more complicated since there are three blocks of measurements (rows) corresponding to the accelerometer measurements, camera measurements and gyroscope measurements, respectively. The parameters (columns) are velocity, acceleration, landmarks, acceleration bias, gyroscope bias, and angular velocity, respectively.

and Montemerlo, 2006; Thrun et al., 2005), is such a method. It is assumed that the data association is known and there is no general mechanism in Graph-SLAM for treating false associations once the graph is constructed. The graph-interpretation is another way of describing dependencies among parameters as in the Jacobian and the Hessian. Examples of two SLAM graphs are shown in Figure 4.3. For instance, the information matrix associated with the Full SLAM posterior is the graph of a *Markov random field* (Thrun et al., 2004). The various graph representations are primarily used to gain insight into the underlying inference problem and to enable efficient solution strategies. For most scenarios the corresponding graph contains loops since some places may be visited several times. Inference on the graph can therefore only be done approximately e.g., with the *sum-product algorithm* or “*loopy*” *belief propagation*. Junction trees eliminates these cycles by clustering them into single nodes (Paskin, 2003) leaving a graph which is a tree and thus exact belief propagation is possible.

The full-SLAM and BA graphs are matrices with known structure when correspondences are fixed. In the tutorial on graph-based SLAM (Grisetti et al., 2011) a clear distinction is made between the sensor specific data association used to construct the graph, which they call the *front-end*, and the strategies used to optimise the graphs which are referred to as the *back-end*. We will here refer to the front-end as the *initialisation*.

4.3.2 Initialisation

Depending on the application the initialisation can be anything from cheap to very computationally expensive. Pose graphs are intuitive to use when the sensor model is invertible which is the case with ground robots having laser range scanners in 2D or 3D. By pair-wise matching of the laser scans, e.g., using *iterative closest point* (ICP), see for instance Besl and McKay (1992), local relative poses are obtained. The pose graph nodes are then the trajectory of robot poses and the edges contains observations and odometry. There are also no natural 3D-3D point correspondences when using laser measurements only since measurements do not provide information around the point. Initialisation in visual applications typically consists of feature tracking and essential matrix estimation within a RANSAC loop.

Figure 4.4: Iterative correspondence search and graph optimisation.

There are few (if any) visual methods that use pose graphs only. For instance, Kim and Eustice (2009); Eade et al. (2010) use view-based matching for whole images combined with odometry and the relative poses obtained are used to construct a pose graph. Both of them are using view-based recognition based on matching entire frames to each other, similar to ICP, thus they are not doing any local tracking. Strasdat et al. (2011) combines a sub-mapping BA approach much like, PTAM (Klein and Murray, 2007), and connects them via a pose graph in a separate optimisation thread. It is good to incorporate iterative an correspondence search in the optimisation thread, as illustrated in Figure 4.4, since associations may become too unlikely by some metric once the optimisation has started. One such error metric for visual tracking is the bi-directional error (Kalal et al., 2010) which is the difference between the image coordinate of the forward-track and the end image coordinate of the backward-track (reversed point tracking) which should be small for well-localised features (Hedborg et al., 2011, 2012). This idea

is similar to what is used in the Paper D where iterates between state estimation and correspondence search is done. Along the same line Dellaert et al. (2003) employ simultaneous SFM and correspondence estimation using EM to learn 3D models. While a multimodal pose graph model is devised Pfingsthorn and Birk (2012) optimising correspondences using Gaussian mixtures which admits corrections for poor initialisation.

4.3.3 NLS-SLAM

In state-space formulations of SLAM the map is included in the state vector without process noise

$$x_t = f(x_{t-1}) + w_t, \quad (4.11a)$$

$$m_t = m_{t-1}, \quad (4.11b)$$

$$y_t = h(x_t, m_t) + e_t. \quad (4.11c)$$

This formulation is however equivalent to

$$x_t = f(x_{t-1}) + w_t, \quad (4.12a)$$

$$y_t = h(x_t, m) + e_t, \quad (4.12b)$$

meaning that the map can be excluded from state vector and simply viewed as a parameter. The idea with NLS-SLAM to use an initial estimate of the map and the whole state-space sequence $x_{0:t}$ and then minimise all the measurement errors and the state trajectory errors as the NLS problem

$$\{\hat{x}_{0:t}, \hat{m}\} = \arg \min_{x_{0:t}, m} \|\bar{x}_0 - x_0\|_{p-1}^2 + \sum_{i=1}^t \|x_i - f(x_{i-1})\|_{Q-1}^2 + \sum_{i=1}^t \|y_i - h(x_i, m)\|_{R-1}^2. \quad (4.13)$$

In Paper D, the dynamic model is considered being exact, i.e., it is just used to define a static map for the parameter-space and is left out of the optimisation. Then the following NLS problem is obtained

$$\{\hat{x}_{0:t}, \hat{m}\} = \arg \min_{x_{0:t}, m} \sum_{i=0}^t \|y_i^m - h_m(x_i, m)\|_{R_m^{-1}}^2 + \|y_i^a - h_a(x_i)\|_{R_a^{-1}}^2 + \|y_i^\omega - h_\omega(x_i)\|_{R_\omega^{-1}}^2, \quad (4.14)$$

where h_m denotes the direct parametrisation (2.47) for camera, h_a denotes accelerometer measurement function and h_ω denotes gyroscope measurements. Besides the IMU measurements, this is a standard BA formulation. Since both the camera and accelerometer measurement functions are nonlinear w.r.t., the state and the map, a good initial value is needed. It is the sole purpose of Paper D to show how such an initial point can be obtained through a sequence of almost linear steps with only SIFT features and IMU data as input.

4.3.4 EM-SLAM

In Paper E the map is also considered being a static parameter, $\theta := m$, according to

$$x_t = f(x_{t-1}, u_t, w_t), \quad (4.15a)$$

$$y_t = h_t(x_t, \theta) + e_t, \quad (4.15b)$$

where the IMU is input to the motion model. In an ML setting, the joint likelihood of measurement and states parametrising the map is

$$p_\theta(y_{1:t}, x_{1:t}) = \prod_{i=1}^t p_\theta(y_i|x_i)p(x_i|x_{i-1}), \quad (4.16)$$

where the state trajectory is considered being a latent variable. This density can be maximised using expectation-maximisation (Dempster et al., 1977) by solving two coupled, but hopefully easier, problems iteratively. This is done by computing the expected value of the log of (4.16)

$$\mathcal{Q}(\theta, \theta_k) = \mathbf{E}_{\theta_k} \left\{ \log \left[\prod_{i=1}^t p_\theta(y_i|x_i)p(x_i|x_{i-1}) \right] \middle| y_{1:t} \right\} = \quad (4.17)$$

$$= - \sum_{i=1}^N \mathbf{E}_{\theta_k} \left\{ \frac{1}{2} \|y_i - h_t(x_i, \theta)\|_{R_m^{-1}}^2 \middle| y_{1:t} \right\} + \text{const.}, \quad (4.18)$$

where the measurement errors are assumed Gaussian. The motion model and other terms independent of the map are lumped into a constant parameter. The expected value cannot be obtained in closed form, instead we approximate (4.17) as

$$\begin{aligned} \mathcal{Q}(\theta, \theta_k) \approx & -\frac{1}{2} \sum_{i=1}^N \left(\|y_i - h_i(\hat{x}_{i|t}, \theta)\|_{R_m^{-1}}^2 + \right. \\ & \left. \text{Tr}(R^{-1} \nabla_x h_i(\hat{x}_{i|t}, \theta) P_{i|t}^s (\nabla_x h_t(\hat{x}_{i|t}, \theta))^T) \right), \end{aligned} \quad (4.19)$$

where, ∇_x denotes the Jacobian w.r.t., to x , $\hat{x}_{i|t}$ is the smoothed estimate of the latent variable and $P_{i|t}^s$ is its covariance. The smoothed estimate is obtained with an E-RTS smoother. The trace term compensates for the use of the estimated latent variables instead of the true ones. The $\mathcal{Q}(\theta, \theta_k)$ function is then maximised w.r.t., the map θ using the quasi-Newton method BFGS, see e.g., Nocedal and Wright (2006) as further explained in Section 3.2 in Paper E. To start the EM-SLAM iterations an initial estimate of the map and the state is needed and it is here obtained using the results in Paper D. It should be noted that the splitting of batch SLAM into state estimation and mapping is also the key in GraphSLAM and FastSLAM as discussed in Section 4.1.1.

4.4 IMU and Camera

Fusion of IMU and monocular vision information is conceptually straightforward but can be a challenging task in practice. A nice introduction to this sensor setup is given in (Corke et al., 2007) and the thorough exposition in Hol (2011) is also recommended. The complementary characteristics of these sensors are attractive since the unbounded error growth in position and orientation from IMU integration can be corrected using the camera.

Figure 4.5: Bearings-only SLAM example illustrating the depth ambiguity. A similar example can also be found in Bailey (2003) but here only the initial velocity 0.25m/s of the platform is known. Left: Landmarks are initialised at approximately the true range and the resulting estimate is consistent. Right: The landmarks are initialised at two times the true range and the estimated trajectory scales accordingly giving a biased estimate. This corresponds to the velocity estimate of the platform being 0.5m/s.

A calibrated stereo camera gives the ability to directly measure 3D coordinates of landmarks in metrical space. However, the baseline between cameras in a stereo rig are physically limited, thus the depth resolution is often limited to close range scenarios since the range resolution itself decreases with range. Motion-stereo does not have this limitation since the baseline is created arbitrary by motion. Obvious downsides of motion-stereo is the need for recovery of the pose, e.g., using epipolar geometry, and the inability to estimate the scale which is due to the depth ambiguity which is illustrated in Figure 4.5. A comparison of stereo and monocular vision based SLAM approaches are presented in Lemaire et al. (2007) and the introductory chapters in Chli (2009) gives a nice overview of vision based SLAM in general.

Apart from SLAM applications the combination of an IMU and monocular camera can be used for:

- IMU supported *structure from motion* (SFM);
- vision supported inertial navigation, i.e., ego-motion estimation;

- loosely or tightly coupled SFM;
- camera IMU fusion for augmented realitys and;
- image stabilisation.

In the context of SLAM, observability is a non-negligible issue and some examples are the case of constant velocity Bryson and Sukkarieh (2008), motion along the optical axis or if landmarks are far away. Short term unobservability is not problematic in filtering problems but may lead to complete failure in batched solutions due to rank deficiency. Simple means for avoiding some of these problems are to use measurement counters. It is however non-trivial to characterise the unobservable subspace in SLAM completely which the substantial amount of publications on this issue is a proof of, see e.g., (Lupton and Sukkarieh, 2012; Hesch et al., 2013; Perera et al., 2009; Kim and Sukkarieh, 2004; Andrade-Cetto and Sanfeliu, 2005; Lee et al., 2006; Wang and Dissanayake, 2008) and the many references therein. Observability is also highlighted in the vision aided IMU systems of Mourikis and Roumeliotis (2007); Mourikis et al. (2009); Li and Mourikis (2013, 2012a); Dong-Si and Mourikis (2012); Li and Mourikis (2012b) which operates without explicit landmark parametrisation and instead keeps old poses in a sliding window expressed as constraints through shared observations.

4.4.1 Linear Triangulation

A monocular camera in combination with an IMU can be used to estimate the pose and to recover the metric scale, a topic which have studied by Martinelli (2012); Nützi et al. (2011); Kneip et al. (2011) and others. It was showed in Martinelli (2012) that given measurements of a landmark from five distinct vantage points with known rotation it is possible to recover the landmark position, the camera position and velocity, and accelerometer bias in closed form. His results are based on well-known linear formulations of the direct parametrisation (2.48) but without considering noise. This formulation was used in Paper D including noise terms and will now be explained. Camera coordinates and the pinhole projection are given by

$$\mathbf{m}^c = \mathbf{R}^{cw}(\mathbf{m}^w - \mathbf{c}_t^w), \quad (4.20a)$$

$$P(\mathbf{m}^c) = \frac{1}{z^c} \begin{bmatrix} x^c \\ y^c \end{bmatrix}, \quad (4.20b)$$

which also can be written as a linear form. With normalised measurements $[u, v]^T$ we have

$$\begin{bmatrix} u \\ v \end{bmatrix} z^c = \begin{bmatrix} x^c \\ y^c \end{bmatrix}, \quad (4.21)$$

which is linear in the unknown $\mathbf{m}^c$ and also in $\mathbf{m}^w$ and $\mathbf{c}_t^w$ if the rotation is known. We can solve (4.21) w.r.t., camera and landmark coordinates with enough measurements available. However, for each new 2D camera measurement there is also another unknown 3D camera position. A solution is to use approximate positions calculated from the accelerometer. This results in a linear system in the

unknown $\mathbf{m}^w$ and we need at least two camera measurement from distinct vantage points to find a solution. In this linear formulation, the camera measurement noise does not reflect depth uncertainty in a proper manner since we have

$$\begin{bmatrix} u \\ v \end{bmatrix} z^c = \begin{bmatrix} x^c \\ y^c \end{bmatrix} + z^c \begin{bmatrix} e_u \\ e_v \end{bmatrix} \iff \begin{bmatrix} u - e_u \\ v - e_v \end{bmatrix} z^c = \begin{bmatrix} x^c \\ y^c \end{bmatrix}, \quad (4.22)$$

meaning that the noise scales with the depth parameter. This can be treated using *iteratively reweighted least squares* (IRWLS), see e.g., Björck (1996) by updating the camera measurement covariance weighting matrix at each iterate using $\hat{z}^c$ from the previous iterate. The key is that we bypass solving the nonlinear re-projection error minimisation problem by solving a simple linear problem. This method is algebraic in the sense that it does not consider minimisation of the measurement errors since the projective model is not directly used to form the error. However, the linear solution also minimise the re-projection error if the correct weighting is used (Zhang and Kanade, 1998; Hartley and Zisserman, 2004) which here corresponds to that $\hat{z}^c$ is sufficiently close to its true value. This section finishes with an example of IRWLS triangulation, inspired by the paper (Hartley and Sturm, 1997).

Example 4.1: IRWLS Triangulation

Consider two 3×4 camera matrices, P and P' with known rotation and translation both measuring a 3D point $\mathbf{m}$. The two cameras and the point are then related by

$$\lambda[u, v, 1]^T = P[\mathbf{m}^T 1]^T = Px, \quad (4.23a)$$

$$\lambda'[u', v', 1]^T = P'[\mathbf{m}^T 1]^T = P'x. \quad (4.23b)$$

The λ 's are unknown scalars accounting for the projective ambiguity. They can be eliminated from the last rows of the two equation systems. After some algebra the unknown 3D point can be expressed as a linear system $Ax = 0$ where $A = [p_1 - up_3, p_2 - vp_3, p'_1 - u'p'_3, p'_2 - v'p'_3]^T$ is a 4×4 matrix and the vectors, $p_i, p'_i, i = 1, 2, 3$, are the rows of the camera matrices. This equation system is solved by computing the SVD of A from which the point is recovered from the singular vector $\sigma[\hat{\mathbf{m}}^T, 1]$ which corresponds to the smallest singular value σ . More measurements can be added as rows in A to further suppress noise.

The minimum of $\|Ax\|$ does not have a geometric meaning. At the minimum there will be an error $\epsilon = up_3^T - p_1^T x$. The re-projection error of the measured image coordinate u and x is given by $\epsilon' = u - p_1^T x / xp_3^T = \epsilon / p_3^T$. Thus, if the first row of A was multiplied with $w = 1/p_3^T x$ and similarly for the other rows, then the minimum of the linear method would correspond to the one of the nonlinear re-projection error. This is not possible to do since it requires x to be known. Instead, we can iterate the linear procedure starting with unit weights and then use the weights from the previous iteration and hopefully converge to a solution with an error close to the one of the re-projection error. The algorithm can for instance be terminated when the change in the weights is small.

The setup is the following: The point is located in $\mathbf{m}^* = [1, 1, 2]^T$, the first camera

P in the origin, the second camera P' is translated to $\mathbf{p}' = [0, 1, 30]^T$ and both of them without rotation. Each camera receives 10 measurements subject to noise $e \sim \mathcal{N}(0, 0.01I_2)$. Results for $M = 100$ simulations and 3 iterations are shown in Table 4.1 where it is clear that the iterations gives an improvement.

Iteration	0	1	2	3
Average RMSE $[\frac{1}{M} \sqrt{\sum_i^M \ \hat{\mathbf{m}}_i - \mathbf{m}^*\ }]$	0.0274	0.0203	0.0202	0.0202

Table 4.1: Iteratively reweighted least squares triangulation.

The strength of this method is that it is simple to compute and no special initialisation is needed, contrary to nonlinear optimisation methods. A disadvantage is that the method may fail to converge (Hartley and Sturm, 1997) if, for example, the baseline is very short which results in a poorly conditioned system.

5

Concluding remarks

This chapter ends the first part of the thesis which consisted of background material. The work in the whole thesis is summarised here with conclusions of the publications in Part II and suggested directions for future work. For detailed conclusions and future work, the reader is referred to each of the appended publication.

5.1 Conclusions

The common denominator for the problems studied in this thesis is that they deal with various aspects of navigation and mapping in unknown environments.

Calibration of sensor networks is important for the network's detection and tracking capabilities. Underwater sensor positions can be difficult to obtain in fast deployment scenarios and sensors can also move due to currents or a non-rigid seabed. An automatic and inexpensive EKF-SLAM method for underwater sensor network positioning without the need for GNSS was presented. Using only magnetic sensors and a vessel with known magnetic signature the sensor positions and the vessel's route was determined. The expected performance of the method and the network was studied using sensitivity- and CRLB analysis on simulated data. This analysis could also be used for sensor network design.

ROV's cannot utilise GNSS for localisation because these signals are greatly attenuated in water. Due to their limited payload capacity, and in order to have a competitive price, the onboard navigation sensors are relatively cheap. We showed that fusion of a complex hydrodynamic model of the ROV with onboard sensor data can improve the navigation performance and the robustness to sensor failure. Experimental results from sea trials showed that in particular the vehicle

speed can be accurately estimated.

The combination of inertial sensor and optical camera can be used both for navigational and mapping purposes. Experiments and simulations indicated that metrically correct estimates can be obtained. The structure of the batch SLAM problem was exploited and solved with NLS and EM utilising efficient optimisation routines. The batch methods requires a good initial estimate for avoiding convergence to a local minimum. A multistage initialisation procedure for batch SLAM was proposed where a series of almost uncoupled and simple problems were solved.

5.2 Future Work

Underwater sensor networks, and sensor networks in general, are important for monitoring and surveillance. The simulated setup in Paper A can certainly benefit from experimental validation which, however, could be rather expensive. A natural extension is to try NLS-SLAM minimisation of the whole data batch to find out if the results can be improved.

Increased navigation performance and robustness w.r.t., disturbances is always desirable. The ROV model in Paper C is used in conjunction with the onboard sensor for this task. The model for rotational dynamics can be improved, perhaps by using speed dependent damping and stronger coupling in the inertia matrix. These, and other, ideas should first be explored using simulations and then preferably in controlled tank tests.

SLAM in general is a mature and diversified field of research. In the near future perhaps the gap between the robotics community and the computer vision community can be shortened. For this to happen, more work on the similarities between the two areas are needed. For instance, it is not that common to see model-based filters in computer vision applications.

System analysis concepts such as observability, controllability and robustness are often difficult to apply to SLAM systems directly. It would therefore be useful to have easy-to-use tools for SLAM system analysis as to guide design and evaluation on a general basis. For the specific application of SLAM with IMU and vision more work on performance bounds, along the line of Nyqvist and Gustafsson (2013), is desirable.

The coupling between filtering and optimisation is an interesting area for more research. Perhaps the convergence rate of Rao-Blackwellized particle filters could be improved by treating some of the nonlinearities using IEKF-L. Iterated smoothers, such as moving horizon estimation further gives the possibility to handle constraints in a systematic manner.

Bibliography

- J. Andrade-Cetto and A. Sanfeliu. The effects of partial observability when building fully correlated maps. *IEEE Transactions on Robotics*, 21(4):771–777, Aug. 2005. ISSN 1552-3098. doi: 10.1109/TRO.2004.842342.
- M. Axholt, M. Skoglund, S. D. Peterson, M. D. Cooper, T. B. Schön, F. Gustafsson, A. Ynnerman, and S. R. Ellis. Optical see-through head mounted display direct linear transformation calibration robustness in the presence of user alignment noise. In *Proceedings of the 54th Annual Meeting of the Human Factors and Ergonomics Society*, volume 54, pages 2427–2431, San Francisco, CA, USA, 27-1 Sept./Oct. 2010. Human Factors and Ergonomics Society.
- M. Axholt, M. A. Skoglund, S. D. O’Connell, M. D. Cooper, S. R. Ellis, and A. Ynnerman. Parameter estimation variance of the single point active alignment method in optical see-through head mounted display calibration. In *Proceedings of the IEEE Virtual Reality Conference*, pages 27–34, Singapore, Republic of Singapore, Mar. 2011.
- T. Bailey. Constrained initialisation for bearing-only SLAM. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, volume 2, pages 1966–1971, Taipei, Taiwan, 14-19 Sept. 2003. doi: 10.1109/ROBOT.2003.1241882.
- Y. Bar-Shalom and X.-R. Li. *Estimation and tracking : principles, techniques and software*. Artech House, Boston, 1993. ISBN 0-89006-643-4.
- B. Bell and F. Cathey. The iterated Kalman filter update as a Gauss-Newton method. *IEEE Transactions on Automatic Control*, 38(2):294–297, Feb. 1993. ISSN 0018-9286. doi: 10.1109/9.250476.
- D. Bertsekas. Incremental least squares methods and the extended Kalman filter. In *Proceedings of the 33rd IEEE Conference on Decision and Control*, volume 2, pages 1211–1214, Lake Buena Vista, Florida, USA, 14-16 Dec. 1994. doi: 10.1109/CDC.1994.411166.
- P. Besl and N. D. McKay. A method for registration of 3-D shapes. *IEEE Transac-*

- tions on Pattern Analysis and Machine Intelligence, 14(2):239–256, Feb. 1992. ISSN 0162-8828. doi: 10.1109/34.121791.
- C. Bibby and I. Reid. Simultaneous localisation and mapping in dynamic environments (SLAMIDE) with reversible data association. In *Proceedings of Robotics: Science and Systems*, Atlanta, GA, USA, 27–30 June 2007.
- Å. Björck. *Numerical Methods for Least Squares Problems*. SIAM, 1996. ISBN 0-89871-360-9.
- M. Bosse and R. Zlot. Map matching and data association for large-scale two-dimensional laser scan-based SLAM. *International Journal of Robotic Research*, 27(6):667–691, 2008.
- J.-Y. Bouguet. Camera Calibration Toolbox for Matlab. www.vision.caltech.edu/bouguetj/calib_doc/, 2010.
- S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004. URL http://www.stanford.edu/~boyd/bv_cvxbook.pdf.
- S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, Jan. 2011. ISSN 1935-8237. doi: 10.1561/22000000016. URL <http://dx.doi.org/10.1561/22000000016>.
- K. Britting. *Inertial Navigation Systems Analysis*. John Wiley & Sons Inc., New York, USA, 1971.
- M. Bryson and S. Sukkarieh. Observability analysis and active control for airborne SLAM. *Aerospace and Electronic Systems, IEEE Transactions on*, 44(1): 261–280, 2008. ISSN 0018-9251. doi: 10.1109/TAES.2008.4517003.
- M. Bryson and S. Sukkarieh. Architectures for Cooperative Airborne Simultaneous Localisation and Mapping. *Journal of Intelligent and Robotic Systems*, 55(4-5):267–297, 2009. doi: 10.1007/s10846-008-9303-9. URL <http://dx.doi.org/10.1007/s10846-008-9303-9>.
- M. Bryson, A. Reid, F. Ramos, and S. Sukkarieh. Airborne vision-based mapping and classification of large farmland environments. *Journal of Field Robotics*, 27(5):632–655, Sept. 2010. ISSN 1556-4959. doi: 10.1002/rob.v27:5. URL <http://dx.doi.org/10.1002/rob.v27:5>.
- F. Caballero, L. Merino, J. Ferruz, and A. Ollero. Vision-based odometry and SLAM for medium and high altitude flying UAVs. *Journal of Intelligent and Robotics Systems*, 54(1-3):137–161, 2009. ISSN 0921-0296. doi: <http://dx.doi.org/10.1007/s10846-008-9257-y>.
- J. Callmer, K. Granström, J. Nieto, and F. Ramos. Tree of words for visual loop closure detection in urban SLAM. In J. Kim and R. Mahony, editors, *Proceedings of the 2008 Australasian Conference on Robotics & Automation*, Canberra, Australia, 3–5 Dec. 2008.

- J. Callmer, M. Skoglund, and F. Gustafsson. Silent localization of underwater sensors using magnetometers. *EURASIP Journal on Advances in Signal Processing*, 2010, 2010. doi: 10.1155/2010/709318. URL <http://dx.doi.org/10.1155/2010/709318>. Article ID 709318.
- J. Callmer, D. Törnqvist, F. Gustafsson, H. Svensson, and P. Carlbom. RADAR SLAM using visual features. *EURASIP Journal on Advances in Signal Processing*, 2011(71), 2011.
- M. Chli. *Applying Information Theory to Efficient SLAM*. PhD thesis, Imperial College London, 2009.
- J. Choi, S. Ahn, and W. K. Chung. Robust sonar feature detection for the SLAM of mobile robot. In *Proceedings of the IEEE/RSJ Conference on Intelligent Robots and Systems*, pages 3415–3420, Edmonton, Alberta, Canada, 2–6 Aug. 2005. doi: 10.1109/IROS.2005.1545284.
- P. Corke, J. Lobo, and J. Dias. An introduction to inertial and visual sensing. *The International Journal of Robotics*, 26:519–535, 2007.
- M. Cummins and P. Newman. Appearance-only SLAM at large scale with FAB-MAP 2.0. *The International Journal of Robotics Research*, 2010. doi: 10.1177/0278364910385483. URL <http://ijr.sagepub.com/content/early/2010/11/11/0278364910385483.abstract>.
- A. J. Davison. Real-time simultaneous localisation and mapping with a single camera. In *Proceedings of the 9th IEEE International Conference on computer vision*, pages 1403–1410, Nice, France, 13–16 Oct. 2003.
- A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse. MonoSLAM: Real-time single camera SLAM. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6):1052–1067, 2007.
- F. Dellaert and M. Kaess. Square Root SAM: Simultaneous location and mapping via square root information smoothing. *International Journal of Robotics Research (IJRR)*, 25(12):1181–1203, 2006. URL <http://www.cc.gatech.edu/~dellaert/pubs/Dellaert06ijrr.pdf>. Special issue on RSS 2006.
- F. Dellaert, S. Seitz, C. Thorpe, and S. Thrun. EM, MCMC, and chain flipping for structure from motion with unknown correspondence. *Machine Learning*, 50 (1-2):45–71, 2003.
- A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977. ISSN 00359246. doi: 10.2307/2984875. URL <http://web.mit.edu/6.435/www/Dempster77.pdf>.
- J. Dennis and R. Schnabel. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Prentice Hall, 1983.
- T. Dong-Si and A. I. Mourikis. Initialization in vision-aided inertial navigation with unknown camera-imu calibration. In *Proceedings of the IEEE/RSJ Inter-*

- national Conference on Robotics and Intelligent Systems (IROS)*, pages 1064–1071, Vilamoura, Portugal, October 2012.
- H. Durrant-Whyte and T. Bailey. Simultaneous Localization and Mapping: Part I. *IEEE Robotics and Automation Magazine*, 13(12):99–110, June 2006.
- E. Eade. *Monocular Simultaneous Localisation and Mapping*. PhD thesis, Cambridge University, 2008.
- E. Eade, P. Fong, and M. Munich. Monocular graph SLAM with complexity reduction. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, Taipei, Taiwan, 18–22 Oct. 2010.
- R. Eustice, H. Singh, J. Leonard, and M. Walter. Visually mapping the RMS Titanic: Conservative covariance estimates for SLAM information filters. *International Journal of Robotics Research*, 25(12):1223–1242, December 2006.
- M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, June 1981. ISSN 0001-0782. doi: 10.1145/358669.358692. URL <http://doi.acm.org/10.1145/358669.358692>.
- R. A. Fisher. On an absolute criterion for fitting frequency curves. *Messenger of Mathematics*, 41:155–160, 1912.
- J. Fossel, D. Hennes, D. Claes, S. Alers, and K. Tuyls. OctoSLAM: A 3D mapping approach to situational awareness of unmanned aerial vehicles. In *Unmanned Aircraft Systems (ICUAS), 2013 International Conference on*, pages 179–188, May 2013. doi: 10.1109/ICUAS.2013.6564688.
- T. I. Fossen. *Handbook of Marine Craft Hydrodynamics and Motion Control*. John Wiley & Sons Ltd., 2011. ISBN 978-1-1199-9149-6.
- F. Fraundorfer and D. Scaramuzza. Visual odometry: Part II: Matching, robustness, optimization, and applications. *IEEE Robotics and Automation Magazine*, 19(2):78–90, 2012. ISSN 1070-9932. doi: 10.1109/MRA.2012.2182810.
- F. Gerossier, P. Checchin, C. Blanc, R. Chapuis, and L. Trassoudaine. Trajectory-oriented EKF-SLAM using the Fourier-Mellin transform applied to microwave radar images. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4925–4930, St. Louis, MO, USA, 11–15 Oct. 2009. doi: 10.1109/IROS.2009.5354548.
- G. Grisetti, R. Kümmerle, C. Stachniss, and W. Burgard. A tutorial on graph-based SLAM. 2(4):31–43, 2011. ISSN 1939-1390. doi: 10.1109/MITS.2010.939925.
- F. Gustafsson. *Statistical Sensor Fusion*. Utbildningshuset/Studentlitteratur, 2012. ISBN 9789144077321.
- S. W. Hamilton. On quaternions; or on a new system of imaginaries in algebra. *Philosophical Magazine*, xxv:10–13, July 1844.

- J. Han and J. Kim. Navigation of an unmanned surface vessel under bridges. In *Proceedings of the International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, pages 206–210, Jeju, Korea, 3-2 Oct./Nov. 2013. doi: 10.1109/URAI.2013.6677343.
- C. Harris and M. Stephens. A combined corner and edge detector. In *proceedings of the 4th Alvey Conference*, pages 147–151, Manchester, UK, 1988.
- R. I. Hartley. Chirality. *International Journal of Computer Vision*, 26(1):41–61, 1998.
- R. I. Hartley and P. Sturm. Triangulation. *Computer Vision and Image Understanding*, 68(2):146 – 157, 1997. ISSN 1077-3142. doi: 10.1006/cviu.1997.0547. URL <http://www.sciencedirect.com/science/article/pii/S1077314297905476>.
- R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2004. ISBN 0-521-54051-8.
- T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer-Verlag, 2 edition, 2009.
- J. Hedborg, E. Ringaby, P.-E. Forssen, and M. Felsberg. Structure and motion estimation from rolling shutter video. In *IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pages 17–23, Barcelona, Spain, Nov. 2011. doi: 10.1109/ICCVW.2011.6130217.
- J. Hedborg, P.-E. Forssén, M. Felsberg, and E. Ringaby. Rolling shutter bundle adjustment. In *IEEE Conference on Computer Vision and Pattern Recognition*, Providence, Rhode Island, USA, June 2012. ISBN 978-1-4673-1227-1. <http://dx.doi.org/10.1109/CVPR.2012.6247831>.
- J. A. Hesch, D. G. Kottas, S. L. Bowman, and S. I. Roumeliotis. Camera-IMU-based localization: Observability analysis and consistency improvement. *The International Journal of Robotics Research*, 2013. doi: 10.1177/0278364913509675. URL <http://ijr.sagepub.com/content/early/2013/11/13/0278364913509675.abstract>.
- J. Hol. *Sensor Fusion and Calibration of Inertial Sensors, Vision, Ultra-Wideband and GPS*. Linköping studies in science and technology. Dissertations. no. 1368, Linköping University, The Institute of Technology, June 2011.
- H. Hopf. Systeme symmetrischer Bilinearformen und euklidische Modelle der projektiven Räume. *Vierteljahrsschrift der Naturforschenden Gesellschaft in Zürich*, 85(Beiblatt (Festschrift Rudolf Fueter)):165–177, 1940.
- A. H. Jazwinski. *Stochastic Processes and Filtering Theory*, volume 64 of *Mathematics in Science and Engineering*. Academic Press, Inc, 1970.
- K. Jönsson. Position Estimation of Remotely Operated Underwater Vehicle. Master’s thesis, Linköping University, 2010.

- I.-K. Jung and S. Lacroix. Simultaneous localization and mapping with stereovision. In *Proceedings of The Eleventh International Symposium Robotics Research*, volume 15 of *Springer Tracts in Advanced Robotics*, pages 315–324, Siena, Italy, 19–22 Oct. 2003. Springer. ISBN 978-3-540-23214-8.
- S.-H. Jung and C. Taylor. Camera trajectory estimation using inertial sensor measurements and structure from motion results. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 2, pages II–732–II–737 vol.2, 2001. doi: 10.1109/CVPR.2001.991037.
- T. Kailath, A. H. Sayed, and B. Hassibi. *Linear Estimation*. Prentice-Hall, Upper Saddle River, New Jersey, 2000.
- Z. Kalal, K. Mikolajczyk, and J. Matas. Forward-backward error: Automatic detection of tracking failures. In *Proceedings of the International Conference on Pattern Recognition (ICPR)*, pages 2756–2759, Istanbul, Turkey, 23–26 Aug. 2010. doi: 10.1109/ICPR.2010.675.
- A. Karlsson and J. Bjärkefur. Simultaneous Localisation and Mapping of Indoor Environments Using a Stereo Camera and a Laser Camera. Master’s thesis, Department of Electrical Engineering, Linköping University, Linköping, Sweden, 2010.
- R. Karlsson, T. Schön, D. Törnqvist, G. Conte, and F. Gustafsson. Utilizing model structure for efficient simultaneous localization and mapping for a UAV application. In *Proceedings of the IEEE Aerospace Conference, Big Sky, MT, USA*, Mar. 2008.
- A. Kim and R. Eustice. Pose-graph visual SLAM with geometric model selection for autonomous underwater ship hull inspection. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1559–1565, St. Louis, MO, USA, 11–15 Oct. 2009. doi: 10.1109/IROS.2009.5354132.
- J. Kim and S. Sukkarieh. Improving the real-time efficiency of inertial SLAM and understanding its observability. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, volume 1, pages 21–26, Sendai, Japan, 28–2 Sept./Oct. 2004. doi: 10.1109/IROS.2004.1389323.
- R. Kim, Ayongand Eustice. Real-time visual SLAM for autonomous underwater hull inspection using visual saliency. *IEEE Transactions on Robotics*, 29(3): 719–733, June 2013. ISSN 1552-3098. doi: 10.1109/TRO.2012.2235699.
- G. Klein and D. Murray. Parallel tracking and mapping for small AR workspaces. In *Proceedings of the International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 225–234, Nara, Japan, 2007.
- L. Kneip, A. Martinelli, S. Weiss, D. Scaramuzza, and R. Siegwart. Closed-form solution for absolute scale velocity determination combining inertial measurements and a single feature correspondence. In *Proceedings of the International*

- Conference on Robotics and Automation*, Shanghai, China, May 2011. URL <http://hal.inria.fr/hal-00641772>.
- K. Konolige and M. Agrawal. FrameSLAM: From bundle adjustment to real-time visual mapping. *IEEE Transactions on Robotics*, 24(5):1066–1077, 2008.
- J. Kuipers. *Quaternions and Rotations Sequences*. Princeton University Press, 2002.
- K. W. Lee, W. Wijesoma, and I. Javier. On the observability and observability analysis of SLAM. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3569–3574, Beijing, China, 9-15 Oct. 2006. doi: 10.1109/IROS.2006.281646.
- T. Lemaire, C. Berger, I.-K. Jung, and S. Lacroix. Vision-based SLAM: Stereo and monocular approaches. *International Journal of Computer Vision*, 74(3):343–364, Sept. 2007. ISSN 0920-5691. doi: 10.1007/s11263-007-0042-3. URL <http://dx.doi.org/10.1007/s11263-007-0042-3>.
- H. Li and R. Hartley. Five-point motion estimation made easy. In *Proceedings of the International Conference on Pattern Recognition*, volume 1, pages 630–633, Hong Kong, 20-24 Aug. 2006. doi: 10.1109/ICPR.2006.579.
- M. Li and A. I. Mourikis. Improving the accuracy of EKF-based visual-inertial odometry. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 828–835, Minneapolis, MN, May 2012a.
- M. Li and A. I. Mourikis. Optimization-based estimator design for vision-aided inertial navigation. In *Proceedings of Robotics: Science and Systems*, Sydney, Australia, July 2012b.
- M. Li and A. I. Mourikis. High-precision, consistent EKF-based visual-inertial odometry. *International Journal of Robotics Research*, 32(6):690–711, May 2013. ISSN 0278-3649. doi: 10.1177/0278364913481251. URL <http://dx.doi.org/10.1177/0278364913481251>.
- H. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135, Sept. 1981.
- D. Lowe. Object Recognition from Local Scale-Invariant Features. In *Proceedings of the Seventh International Conference on Computer Vision (ICCV’99)*, pages 1150–1157, Corfu, Greece, 1999.
- F. Lu and E. Milios. Globally consistent range scan alignment for environment mapping. *Autonomous Robots*, 4:333–349, 1997.
- C. Lundquist, M. A. Skoglund, K. Granström, and T. Glad. Insights from implementing a system for peer-review. *IEEE Transactions on Education*, 56(3): 261–267, 2013.
- T. Lupton and S. Sukkarieh. Removing scale biases and ambiguity from 6DoF monocular SLAM using inertial. In *Proceedings of the International Confer-*

- ence on Robotics and Automation (ICRA), pages 3698–3703, Pasadena, California, USA, 2008. IEEE. doi: 10.1109/ROBOT.2008.4543778. URL <http://dx.doi.org/10.1109/ROBOT.2008.4543778>.
- T. Lupton and S. Sukkarieh. Efficient integration of inertial observations into visual SLAM without initialization. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1547–1552, St. Louis, MO, USA, 11-15 Oct. 2009. IEEE. doi: 10.1109/IROS.2009.5354267. URL <http://dx.doi.org/10.1109/IROS.2009.5354267>.
- T. Lupton and S. Sukkarieh. Visual-inertial-aided navigation for high-dynamic motion in built environments without initial conditions. *IEEE Transactions on Robotics*, 28(1):61–76, Feb. 2012. ISSN 1552-3098. doi: 10.1109/TRO.2011.2170332.
- M. Kaess and A. Ranganathan and F. Dellaert. iSAM: Incremental smoothing and mapping. *IEEE Transactions on Robotics*, 24(6):1365–1378, Dec. 2008.
- Y. Ma, S. Soatto, J. Kosecka, and S. S. Sastry. *An Invitation to 3-D Vision: From Images to Geometric Models*. Springer Verlag, 2003. ISBN 0387008934.
- I. Mahon, S. B. Williams, O. Pizarro, and M. Johnson-Roberson. Efficient view-based SLAM using visual loop closures. *IEEE Transactions on Robotics*, 24(5):1002–1014, 2008.
- J. Marck, A. Mohamoud, E. v.d.Houwen, and R. van Heijster. Indoor radar SLAM a radar application for vision and gps denied environments. In *Proceedings of the European Radar Conference (EuRAD)*.
- D. W. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *SIAM Journal on Applied Mathematics*, 11(2):431–441, 1963. doi: 10.1137/0111030. URL <http://dx.doi.org/10.1137/0111030>.
- A. Martinelli. Vision and imu data fusion: Closed-form solutions for attitude, speed, absolute scale, and bias determination. *IEEE Transactions on Robotics*, 28(1):44–60, Feb. 2012. ISSN 1552-3098. doi: 10.1109/TRO.2011.2160468.
- J. Mattingley and S. Boyd. Real-time convex optimization in signal processing. *IEEE Signal Processing Magazine*, 27(3):50–61, 2010. ISSN 1053-5888. doi: 10.1109/MSP.2010.936020.
- M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. FastSLAM: A factored solution to the simultaneous localization and mapping problem. In *Proceedings of the AAAI National Conference on Artificial Intelligence*, page 593–598, Edmonton, Alberta, Canada, 28-1 July/Aug. 2002.
- M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. FastSLAM 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI)*, Acapulco, Mexico, 9-15 Aug. 2003.

- J. Montiel and A. Davison. A visual compass based on SLAM. In *Proceedings the IEEE International Conference on Robotics and Automation (ICRA)*, pages 1917–1922, Orlando, Florida, USA, May 2006. doi: 10.1109/ROBOT.2006.1641986.
- A. Mourikis and S. Roumeliotis. A multi-state constraint Kalman filter for vision-aided inertial navigation. In *Proceedings of the IEEE International Conference on Robotics and Automation*, Roma, Italy, 10-14 Apr. 2007.
- A. I. Mourikis, N. Trawny, S. I. Roumeliotis, A. E. Johnson, A. Ansar, and L. Matthies. Vision-aided inertial navigation for spacecraft entry, descent, and landing. *IEEE Transactions on Robotics*, 25(2):264 – 280, 2009. ISSN 15523098. URL <https://lt.ltag.bibl.liu.se/login?url=http://search.ebscohost.com/login.aspx?direct=true&db=buh&AN=38716223&site=ehost-live>.
- J. Mullane, S. Keller, A. Rao, M. Adams, A. Yeo, F. Hover, and N. Patrikalakis. X-band radar based SLAM in Singapore’s off-shore environment. In *Proceedings of the International Conference on Control Automation Robotics Vision (ICARCV)*, Singapore, 7-10 Dec. 2010.
- P. M. Newman, J. J. Leonard, and R. J. Rikoski. Towards constant-time SLAM on an autonomous underwater vehicle using synthetic aperture sonar. In *Proceedings of the Eleventh International Symposium on Robotics Research*, pages 409–420, Siena, Italy, 19-22 Oct. 2003. Springer Verlag.
- E. Nilsson. An optimization based approach to visual odometry using infrared images. Master’s thesis, Department of Electrical Engineering, Linköping University, Linköping, Sweden, 2010.
- E. Nilsson, C. Lundquist, T. B. Schön, D. Forslund, and J. Roll. Vehicle motion estimation using an infrared camera. In *Proceedings of the IFAC World Congress*, Milan, Italy, Aug./Sept. 2011.
- D. Nistér. Reconstruction from uncalibrated sequences with a hierarchy of trifocal tensors. In *Proceedings of the 6th European Conference on Computer Vision-Part I, ECCV ’00*, pages 649–663, London, UK, UK, 2000. Springer-Verlag. ISBN 3-540-67685-6. URL <http://dl.acm.org/citation.cfm?id=645313.649415>.
- D. Nistér. An efficient solution to the five-point relative pose problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(6):756–770, 2004. ISSN 0162-8828. doi: 10.1109/TPAMI.2004.17.
- J. Nocedal. Updating quasi-Newton matrices with limited storage. *Mathematics of Computation*, 35(151):773–782, 1980. URL <http://www.jstor.org/stable/2006193>.
- J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, New York, 2nd edition, 2006. ISBN 978-0-387-40065-5.

- G. Nützi, S. Weiss, D. Scaramuzza, and R. Siegwart. Fusion of IMU and vision for absolute scale estimation in monocular SLAM. *Journal of Intelligent Robotics Systems*, 61:287–299, Jan. 2011. ISSN 0921-0296. doi: <http://dx.doi.org/10.1007/s10846-010-9490-z>. URL <http://dx.doi.org/10.1007/s10846-010-9490-z>.
- H. Nyqvist and F. Gustafsson. A high-performance tracking system based on camera and imu. In *Proceedings of the 16th International Conference on Information Fusion*, Istanbul, Turkey, 9-12 July 2013.
- M. Paskin. Thin junction tree filters for simultaneous localization and mapping. In *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence (IJCAI-03)*, pages 1157–1164, San Francisco, CA, 2003.
- L. Perera, A. Melkumyan, and E. Nettleton. On the linear and nonlinear observability analysis of the SLAM problem. pages 1–6, Málaga, Spain, 14-17 Apr. 2009. doi: 10.1109/ICMECH.2009.4957168.
- M. Pfingsthorn and A. Birk. Simultaneous localization and mapping (slam) with multimodal probability distributions. *The International Journal of Robotics Research*, 2012. doi: 10.1177/0278364912461540. URL <http://ijr.sagepub.com/content/early/2012/10/05/0278364912461540.abstract>.
- P. Pinies, T. Lupton, S. Sukkarieh, and J. Tardos. Inertial aiding of inverse depth SLAM using a monocular camera. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2797–2802, Roma, Italy, 10-14 Apr. 2007.
- C. Rao. *Moving Horizon Strategies for the Constrained Monitoring and Control of Nonlinear Discrete-Time Systems*. PhD thesis, University of Wisconsin Madison, 2000.
- H. E. Rauch, S. C. T., and T. F. Maximum likelihood estimates of linear dynamic systems. *AIAA Journal*, 3(8):1445–1450, 1965.
- D. Ribas, P. Ridao, J. Neira, and J. Tardos. SLAM using an imaging sonar for partially structured underwater environments. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Beijing, China, 9-15 Oct. 2006.
- D. Scaramuzza and F. Fraundorfer. Visual odometry : Part I: The first 30 years and fundamentals. *IEEE Robotics and Automation Magazine*, 18(4):80–92, 2011. ISSN 1070-9932. doi: 10.1109/MRA.2011.943233.
- D. Scaramuzza, A. Martinelli, and R. Siegwart. A toolbox for easily calibrating omnidirectional cameras. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Beijing, China, 9-15 Oct. 2006.
- J. Shi and C. Tomasi. Good features to track. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*,

- pages 593–600, Seattle, WA, USA, 21–23 June 1994. doi: 10.1109/CVPR.1994.323794.
- M. D. Shuster. Survey of attitude representations. *Journal of the Astronautical Sciences*, 41(4):439–517, 1993.
- M. D. Shuster. Constraint in attitude estimation part II: Unconstrained estimation. *The Journal of the Astronautical Sciences*, 51(1):75–101, Jan.–Mar. 2003.
- Z. Sjanic and F. Gustafsson. Simultaneous navigation and SAR auto-focusing. In *Proceedings of the International Conference on Information Fusion (FUSION)*, pages 1–7, Edinburgh, UK, 26–29 July 2010.
- Z. Sjanic, M. A. Skoglund, F. Gustafsson, and T. B. Schön. A nonlinear least squares approach to the SLAM problem. In *Proceedings of the IFAC World Congress*, volume 18, Milan, Italy, 28–2 Aug./Sept. 2011.
- Z. Sjanic, M. A. Skoglund, and F. Gustafsson. EM-SLAM with inertial/visual applications. Submitted to IEEE Transactions on Aerospace and Electronic Systems, June 2014.
- M. A. Skoglund, K. Jönsson, and F. Gustafsson. Modeling and sensor fusion of a remotely operated underwater vehicle. In *Proceedings of the 15th International Conference on Information Fusion (FUSION), Singapore, 9–12 July 2012*, pages 947–954. IEEE, 2012.
- M. A. Skoglund, Z. Sjanic, and F. Gustafsson. Initialisation and estimation methods for batch optimisation of inertial/visual SLAM. Submitted to IEEE Transactions on Aerospace and Electronic Systems, June 2014.
- R. Smith, M. Self, and P. Cheeseman. Estimating uncertain spatial relationships in robotics. In *Autonomous robot vehicles*, pages 167–193. Springer-Verlag New York, Inc., New York, NY, USA, 1990. ISBN 0-387-97240-4.
- H. Strasdat, A. Davison, J. M. M. Montiel, and K. Konolige. Double window optimisation for constant time visual SLAM. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Barcelona, Spain, 6–13 Nov. 2011.
- J. Stuelpnagel. On the parametrization of the three-dimensional rotation group. *SIAM Review*, 6(4):pp. 422–430, 1964. ISSN 00361445. URL <http://www.jstor.org/stable/2027966>.
- T. Thormählen, N. Hasler, M. Wand, and H.-P. Seidel. Merging of Feature Tracks for Camera Motion Estimation from Video. In *Proceedings of the 5th European Conference on Visual Media Production (CVMP 2008)*, pages 1 – 8, Hertfordshire, UK, Jan. 2008. IET.
- S. Thrun and M. Montemerlo. The GraphSLAM algorithm with applications to large-scale mapping of urban structures. *International Journal on Robotics Research*, 25(5):403–430, 2006.

- S. Thrun, Y. Liu, D. Koller, A. Ng, Z. Ghahramani, and H. Durrant-Whyte. Simultaneous localization and mapping with sparse extended information filters. *International Journal of Robotics Research*, 23(7/8), 2004.
- S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press, 2005. ISBN 0262201623.
- R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society (Series B)*, 58:267–288, 1996.
- D. Titterton and J. Weston. *Strapdown Inertial Navigation Technology*. IEE radar, sonar, navigation and avionics series. Peter Peregrinus Ltd., Stevenage, UK, 1997. ISBN 0863413587.
- D. Törnqvist. *Estimation and Detection with Applications to Navigation*. Linköping Studies in Science and Technology. Dissertations. No. 1216, Linköping University, Nov. 2008.
- P. Torr and A. Zisserman. MLESAC: A new robust estimator with application to estimating image geometry. *Computer Vision and Image Understanding*, 78(1):138 – 156, 2000. ISSN 1077-3142. doi: 10.1006/cviu.1999.0832. URL <http://www.sciencedirect.com/science/article/pii/S1077314299908329>.
- U. Frese. Treemap: An $\mathcal{O}(\log n)$ algorithm for simultaneous localization and mapping. In C. Freksa, editor, *Spatial Cognition IV*, pages 455–476. Springer Verlag, 2005. URL http://www.informatik.uni-bremen.de/~ufrese/slamvideos1_e.html.
- N. Wahlström. *Localization using Magnetometers and Light Sensors*. Linköping Studies in Science and Technology. Thesis No. 1581, Linköping University, Sweden, 2013.
- Z. Wang and G. Dissanayake. Observability analysis of SLAM using Fisher information matrix. In *Proceedings of the International Conference on Control, Automation, Robotics and Vision (ICARCV)*, pages 1242–1247, Hanoi, Vietnam, 17-20 Dec. 2008. doi: 10.1109/ICARCV.2008.4795699.
- Z. Wang and G. Dissanayake. Exploiting vehicle motion information in monocular SLAM. In *Proceedings of the International Conference on Control Automation Robotics Vision (ICARCV)*, pages 1030–1035, Guangzhou, China, 5-7 Dec. 2012. doi: 10.1109/ICARCV.2012.6485299.
- W. Wijesoma, L. Perera, and M. Adams. Toward multidimensional assignment data association in robot localization and mapping. *IEEE Transactions on Robotics*, 22(2):350 – 365, Apr. 2006. ISSN 1552-3098. doi: 10.1109/TRO.2006.870634.
- K. Wrobel. SLAM-based approach to dynamic ship positioning. *TransNav, the International Journal on Marine Navigation and Safety of Sea Transportation*, 8(1):21–25, 2014.

- Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, Nov. 2000. ISSN 0162-8828. doi: 10.1109/34.888718.
- Z. Zhang and T. Kanade. Determining the epipolar geometry and its uncertainty: A review. *International Journal of Computer Vision*, 27:161–195, 1998.

Part II

Publications

Publications

The articles associated with this thesis have been removed for copyright reasons. For more details about these see:

<http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-110373>

P. Pucar: Modeling and segmentation using multiple models. Thesis No. 405, 1995. ISBN 91-7871-627-6.

H. Fortell: Algebraic approaches to normal forms and zero dynamics. Thesis No. 407, 1995. ISBN 91-7871-629-2.

A. Helmersson: Methods for robust gain scheduling. Thesis No. 406, 1995. ISBN 91-7871-628-4.

P. Lindskog: Methods, algorithms and tools for system identification based on prior knowledge. Thesis No. 436, 1996. ISBN 91-7871-424-8.

J. Gunnarsson: Symbolic methods and tools for discrete event dynamic systems. Thesis No. 477, 1997. ISBN 91-7871-917-8.

M. Jirstrand: Constructive methods for inequality constraints in control. Thesis No. 527, 1998. ISBN 91-7219-187-2.

U. Forssell: Closed-loop identification: Methods, theory, and applications. Thesis No. 566, 1999. ISBN 91-7219-432-4.

A. Stenman: Model on demand: Algorithms, analysis and applications. Thesis No. 571, 1999. ISBN 91-7219-450-2.

N. Bergman: Recursive Bayesian estimation: Navigation and tracking applications. Thesis No. 579, 1999. ISBN 91-7219-473-1.

K. Edström: Switched bond graphs: Simulation and analysis. Thesis No. 586, 1999. ISBN 91-7219-493-6.

M. Larsson: Behavioral and structural model based approaches to discrete diagnosis. Thesis No. 608, 1999. ISBN 91-7219-615-5.

F. Gunnarsson: Power control in cellular radio systems: Analysis, design and estimation. Thesis No. 623, 2000. ISBN 91-7219-689-0.

V. Einarsson: Model checking methods for mode switching systems. Thesis No. 652, 2000. ISBN 91-7219-836-2.

M. Norrlöf: Iterative learning control: Analysis, design, and experiments. Thesis No. 653, 2000. ISBN 91-7219-837-0.

F. Tjärnström: Variance expressions and model reduction in system identification. Thesis No. 730, 2002. ISBN 91-7373-253-2.

J. Löfberg: Minimax approaches to robust model predictive control. Thesis No. 812, 2003. ISBN 91-7373-622-8.

J. Roll: Local and piecewise affine approaches to system identification. Thesis No. 802, 2003. ISBN 91-7373-608-2.

J. Elbornsson: Analysis, estimation and compensation of mismatch effects in A/D converters. Thesis No. 811, 2003. ISBN 91-7373-621-X.

O. Härkegård: Backstepping and control allocation with applications to flight control. Thesis No. 820, 2003. ISBN 91-7373-647-3.

R. Wallin: Optimization algorithms for system analysis and identification. Thesis No. 919, 2004. ISBN 91-85297-19-4.

D. Lindgren: Projection methods for classification and identification. Thesis No. 915, 2005. ISBN 91-85297-06-2.

R. Karlsson: Particle Filtering for Positioning and Tracking Applications. Thesis No. 924, 2005. ISBN 91-85297-34-8.

J. Jansson: Collision Avoidance Theory with Applications to Automotive Collision Mitigation. Thesis No. 950, 2005. ISBN 91-85299-45-6.

E. Geijer Lundin: Uplink Load in CDMA Cellular Radio Systems. Thesis No. 977, 2005. ISBN 91-85457-49-3.

M. Enqvist: Linear Models of Nonlinear Systems. Thesis No. 985, 2005. ISBN 91-85457-64-7.

T. B. Schön: Estimation of Nonlinear Dynamic Systems — Theory and Applications. Thesis No. 998, 2006. ISBN 91-85497-03-7.

I. Lind: Regressor and Structure Selection — Uses of ANOVA in System Identification. Thesis No. 1012, 2006. ISBN 91-85523-98-4.

J. Gillberg: Frequency Domain Identification of Continuous-Time Systems Reconstruction and Robustness. Thesis No. 1031, 2006. ISBN 91-85523-34-8.

M. Gerdin: Identification and Estimation for Models Described by Differential-Algebraic Equations. Thesis No. 1046, 2006. ISBN 91-85643-87-4.

C. Grönwall: Ground Object Recognition using Laser Radar Data – Geometric Fitting, Performance Analysis, and Applications. Thesis No. 1055, 2006. ISBN 91-85643-53-X.

A. Eidehall: Tracking and threat assessment for automotive collision avoidance. Thesis No. 1066, 2007. ISBN 91-85643-10-6.

F. Eng: Non-Uniform Sampling in Statistical Signal Processing. Thesis No. 1082, 2007. ISBN 978-91-85715-49-7.

E. Wernholt: Multivariable Frequency-Domain Identification of Industrial Robots. Thesis No. 1138, 2007. ISBN 978-91-85895-72-4.

D. Axehill: Integer Quadratic Programming for Control and Communication. Thesis No. 1158, 2008. ISBN 978-91-85523-03-0.

G. Hendeby: Performance and Implementation Aspects of Nonlinear Filtering. Thesis No. 1161, 2008. ISBN 978-91-7393-979-9.

J. Sjöberg: Optimal Control and Model Reduction of Nonlinear DAE Models. Thesis No. 1166, 2008. ISBN 978-91-7393-964-5.

D. Törnqvist: Estimation and Detection with Applications to Navigation. Thesis No. 1216, 2008. ISBN 978-91-7393-785-6.

P.-J. Nordlund: Efficient Estimation and Detection Methods for Airborne Applications. Thesis No. 1231, 2008. ISBN 978-91-7393-720-7.

H. Tidefelt: Differential-algebraic equations and matrix-valued singular perturbation. Thesis No. 1292, 2009. ISBN 978-91-7393-479-4.

H. Ohlsson: Regularization for Sparseness and Smoothness — Applications in System Identification and Signal Processing. Thesis No. 1351, 2010. ISBN 978-91-7393-287-5.

S. Moberg: Modeling and Control of Flexible Manipulators. Thesis No. 1349, 2010. ISBN 978-91-7393-289-9.

J. Wallén: Estimation-based iterative learning control. Thesis No. 1358, 2011. ISBN 978-91-7393-255-4.

J. Hol: Sensor Fusion and Calibration of Inertial Sensors, Vision, Ultra-Wideband and GPS. Thesis No. 1368, 2011. ISBN 978-91-7393-197-7.

D. Ankelhed: On the Design of Low Order H-infinity Controllers. Thesis No. 1371, 2011. ISBN 978-91-7393-157-1.

C. Lundquist: Sensor Fusion for Automotive Applications. Thesis No. 1409, 2011. ISBN 978-91-7393-023-9.

P. Skoglar: Tracking and Planning for Surveillance Applications. Thesis No. 1432, 2012. ISBN 978-91-7519-941-2.

K. Granström: Extended target tracking using PHD filters. Thesis No. 1476, 2012. ISBN 978-91-7519-796-8.

C. Lyzell: Structural Reformulations in System Identification. Thesis No. 1475, 2012. ISBN 978-91-7519-800-2.

J. Callmer: Autonomous Localization in Unknown Environments. Thesis No. 1520, 2013. ISBN 978-91-7519-620-6.

D. Petersson: A Nonlinear Optimization Approach to H2-Optimal Modeling and Control. Thesis No. 1528, 2013. ISBN 978-91-7519-567-4.

Z. Sjanic: Navigation and Mapping for Aerial Vehicles Based on Inertial and Imaging Sensors. Thesis No. 1533, 2013. ISBN 978-91-7519-553-7.

F. Lindsten: Particle Filters and Markov Chains for Learning of Dynamical Systems. Thesis No. 1530, 2013. ISBN 978-91-7519-559-9.

P. Axelsson: Sensor Fusion and Control Applied to Industrial Manipulators. Thesis No. 1585, 2014. ISBN 978-91-7519-368-7.

A. Carvalho Bittencourt: Modeling and Diagnosis of Friction and Wear in Industrial Robots. Thesis No. 1617, 2014. ISBN 978-91-7519-251-2.