Institut für Geodäsie und Geoinformation

Schriftenreihe

Visual Odometry and Sparse Scene Reconstruction for UAVs with a Multi-Fisheye Camera System

Johannes Schneider

67

Bonn 2020

Johannes Schneider · **Visual Odometry and Sparse Scene Reconstruction for UAVs with a Multi-Fisheye Camera System**

Diese Dissertation zur Erlangung des Grades Doktor der Ingenieurwissenschaften (Dr.-Ing.) wurde an der Landwirtschaftlichen Fakultät der Rheinischen Friedrich-Wilhelms-Universität Bonn vorgelegt.

Referent:	Prof. Dr. Wolfgang Förstner
1. Korreferent:	Prof. Dr. Cyrill Stachniss
2. Korreferent:	Prof. Dr. Christian Heipke

Tag der mündlichen Prüfung: 27. Mai 2019

Diese Dissertation ist auf bonndoc - dem Publikationsserver der Universität Bonn elektronisch publiziert:
https://nbn-resolving.org/urn:nbn:de:hbz:5n-54867

Schriftenreihe des Instituts für Geodäsie und Geoinformation
der Rheinischen Friedrich-Wilhelms-Universität Bonn

Herausgeber: Prof. Dr. Jan-Henrik Haunert
Prof. Dr.-Ing. Theo Kötter
Prof. Dr.-Ing. Heiner Kuhlmann
Prof. Dr.-Ing. Jürgen Kusche
Prof. Dr. techn. Wolf-Dieter Schuh
Prof. Dr. Cyrill Stachniss

Die Aufnahme dieser Arbeit in die Schriftenreihe wurde von den Herausgebern
der Reihe einstimmig beschlossen.

Dieses Werk ist einschließlich aller seiner Teile urheberrechtlich geschützt.
Abdruck auch auszugsweise nur mit Quellenangabe gestattet.
Alle Rechte vorbehalten.

Schriftenreihe Heft 67

Bonn 2020 2699-6685 (Online)

UNIVERSITÄT BONN

Inaugural-Dissertation
zur Erlangung des Grades
Doktor der Ingenieurwissenschaften (Dr.-Ing.)
der Landwirtschaftlichen Fakultät
der Rheinischen Friedrich-Wilhelms-Universität Bonn
Institut für Geodäsie und Geoinformation

Visual Odometry and Sparse Scene Reconstruction for UAVs with a Multi-Fisheye Camera System

von

Johannes Schneider

aus

Kleve, Germany

Bonn 2019

Referent:
Prof. Dr.-Ing. Dr. h.c. mult. Wolfgang Förstner, Friedrich-Wilhelms-Universität Bonn

1. Korreferent:
Prof. Dr. Cyrill Stachniss, Friedrich-Wilhelms-Universität Bonn

2. Korreferent:
Prof. Dr. Christian Heipke, Leibniz Universität Hannover

Tag der mündlichen Prüfung: 27. Mai 2019

Angefertigt mit Genehmigung der Landwirtschaftlichen Fakultät der Universität Bonn

Abstract

Autonomously operating UAVs demand a fast localization for navigation, to actively explore unknown areas and to create maps. For pose estimation, many UAV systems make use of a combination of GPS receivers and inertial sensor units (IMU). However, GPS signal coverage may go down occasionally, especially in the close vicinity of objects, and precise IMUs are too heavy to be carried by lightweight UAVs. This and the high cost of high quality IMU motivate the use of inexpensive vision based sensors for localization using visual odometry or visual SLAM (simultaneous localization and mapping) techniques.

The first contribution of this thesis is a more general approach to bundle adjustment with an extended version of the projective coplanarity equation which enables us to make use of omnidirectional multi-camera systems which may consist of fisheye cameras that can capture a large field of view with one shot. We use ray directions as observations instead of image points which is why our approach does not rely on a specific projection model assuming a central projection. In addition, our approach allows the integration and estimation of points at infinity, which classical bundle adjustments are not capable of. We show that the integration of far or infinitely far points stabilizes the estimation of the rotation angles of the camera poses.

In its second contribution, we employ this approach to bundle adjustment in a highly integrated system for incremental pose estimation and mapping on light-weight UAVs. Based on the image sequences of a multi-camera system our system makes use of tracked feature points to incrementally build a sparse map and incrementally refines this map using the iSAM2 algorithm. Our system is able to optionally integrate GPS information on the level of carrier phase observations even in underconstrained situations, e.g. if only two satellites are visible, for georeferenced pose estimation. This way, we are able to use all available information in underconstrained GPS situations to keep the mapped 3D model accurate and georeferenced.

In its third contribution, we present an approach for re-using existing methods for dense stereo matching with fisheye cameras, which has the advantage that highly optimized existing methods can be applied as a black-box without modifications even with cameras that have field of view of more than 180°. We provide a detailed accuracy analysis of the obtained dense stereo results. The accuracy analysis shows the growing uncertainty of observed image points of fisheye cameras due to increasing blur towards the image border. Core of the contribution is a rigorous variance component estimation which allows to estimate the variance of the observed disparities at an image point as a function of the distance of that point to the principal point. We show that this improved stochastic model provides a more realistic prediction of the uncertainty of the triangulated 3D points.

Zusammenfassung

Autonom operierende UAVs benötigen eine schnelle Lokalisierung zur Navigation, zur Exploration unbekannter Umgebungen und zur Kartierung. Zur Posenbestimmung verwenden viele UAV-Systeme eine Kombination aus GPS-Empfängern und Inertial-Messeinheiten (IMU). Die Verfügbarkeit von GPS-Signalen ist jedoch nicht überall gewährleistet, insbesondere in der Nähe abschattender Objekte, und präzise IMUs sind für leichtgewichtige UAVs zu schwer. Auch die hohen Kosten qualitativ hochwertiger IMUs motivieren den Einsatz von kostengünstigen bildgebenden Sensoren zur Lokalisierung mittels visueller Odometrie oder SLAM-Techniken zur simultanen Lokalisierung und Kartierung.

Im ersten wissenschaftlichen Beitrag dieser Arbeit entwickeln wir einen allgemeineren Ansatz für die Bündelausgleichung mit einem erweiterten Modell für die projektive Kollinearitätsgleichung, sodass auch omnidirektionale Multikamerasysteme verwendet werden können, welche beispielsweise bestehend aus Fisheyekameras mit einer Aufnahme einen großen Sichtbereich abdecken. Durch die Integration von Strahlrichtungen als Beobachtungen ist unser Ansatz nicht von einem kameraspezifischen Abbildungsmodell abhängig solange dieses der Zentralprojektion folgt. Zudem erlaubt unser Ansatz die Integration und Schätzung von unendlich fernen Punkten, was bei klassischen Bündelausgleichungen nicht möglich ist. Wir zeigen, dass durch die Integration weit entfernter und unendlich ferner Punkte die Schätzung der Rotationswinkel der Kameraposen stabilisiert werden kann.

Im zweiten Beitrag verwenden wir diesen entwickelten Ansatz zur Bündelausgleichung für ein System zur inkrementellen Posenschätzung und dünnbesetzten Kartierung auf einem leichtgewichtigen UAV. Basierend auf den Bildsequenzen eines Mulitkamerasystems baut unser System mittels verfolgter markanter Bildpunkte inkrementell eine dünnbesetzte Karte auf und verfeinert diese inkrementell mittels des iSAM2-Algorithmus. Unser System ist in der Lage optional auch GPS Informationen auf dem Level von GPS-Trägerphasen zu integrieren, wodurch sogar in unterbestimmten Situation – beispielsweise bei nur zwei verfügbaren Satelliten – diese Informationen zur georeferenzierten Posenschätzung verwendet werden können.

Im dritten Beitrag stellen wir einen Ansatz zur Verwendung existierender Methoden für dichtes Stereomatching mit Fisheyekameras vor, sodass hoch optimierte existierende Methoden als Black Box ohne Modifzierungen sogar mit Kameras mit einem Gesichtsfeld von mehr als 180° verwendet werden können. Wir stellen eine detaillierte Genauigkeitsanalyse basierend auf dem Ergebnis des dichten Stereomatchings dar. Die Genauigkeitsanalyse zeigt, wie stark die Genauigkeit beobachteter Bildpunkte bei Fisheyekameras zum Bildrand aufgrund von zunehmender Unschärfe abnimmt. Das Kernstück dieses Beitrags ist eine Varianzkomponentenschätzung, welche die Schätzung der Varianz der beobachteten Disparitäten an einem Bildpunkt als Funktion von der Distanz dieses Punktes zum Hauptpunkt des Bildes ermöglicht. Wir zeigen, dass dieses verbesserte stochastische Modell eine realistischere Prädiktion der Genauigkeiten der 3D Punkte ermöglicht.

Contents

1 Introduction 11

1.1 Motivation 12

1.2 The Copter Design 13

1.3 Contribution and Organization 15

1.3.1 Contributions 15

1.3.2 Organization 16

1.4 Publications 17

1.5 Notation 18

2 Basic Techniques 21

2.1 Uncertainty of Homogeneous Representations and Transformations 21

2.1.1 Uncertain Homogeneous Points 22

2.1.2 Uncertain Rotations 23

2.1.3 Uncertain Homogeneous Spatial Motions 24

2.2 Basic Image Geometry 26

2.2.1 Interior Orientation 27

2.2.2 Camera Calibration 30

2.3 Weighted Least-squares Estimation 31

2.3.1 Estimation with Non-linear Gauss–Markov Model 32

2.3.2 Robust Estimation 33

2.4 Incremental Estimation 35

2.4.1 QR Matrix Factorization 35

2.4.2 Incremental Factorization with New Observations and Unknowns . . 36

2.4.3 The iSAM2 Algorithm 37

3 Bundle Adjustment for Multi-Camera Systems with Far Points 41

3.1 Introduction 42

3.1.1 Multi-camera Systems 42

3.1.2 Omnidirectional Cameras 43

3.1.3 Points at Infinity 43

3.1.4 The Idea 45

3.1.5 Task and Challenges 46

3.2 Related Work 47

3.3 Model for a Moving Single-View Camera 49
3.3.1 Image Coordinates as Observations 49
3.3.2 Ray Directions as Observations 49
3.3.3 Handling Far and Ideal Scene Points 50
3.4 Model for a Moving Multi-camera System 51
3.5 Generating Camera Directions from Observed Image Coordinates 51
3.5.1 Perspective Cameras 52
3.5.2 Omnidirectional Single View Point Cameras 53
3.6 The Estimation Procedure 54
3.6.1 Initial Values 54
3.6.2 Linearization and Update for Pose Parameters 55
3.6.3 Reduced Coordinates and Update of Coordinates 56
3.6.4 Linearized Model for Bundle Adjustment 58
3.7 Experiments 60
3.7.1 Implementation Details 60
3.7.2 Test on Correctness and Feasibility 61
3.7.3 Decrease of Rotational Precision Excluding Far Points 63
3.7.4 Calibration of Multi-Camera Systems 66
3.8 Conclusion 71

4 Visual Odometry for Omnidirectional Camera Systems 73
4.1 Introduction 73
4.2 Related Work 74
4.3 Online Pose Estimation and Mapping 76
4.3.1 Overview 76
4.3.2 Visual Data Acquisition and Association 77
4.3.3 Fast Pose Estimation 78
4.3.4 Keyframe-Based Incremental Bundle Adjustment 79
4.3.5 Integration of GPS and IMU Information 81
4.4 Experiments 84
4.4.1 Real-time Capabilities and Optimality of Incremental Bundle Adjustment 85
4.4.2 Localization Precision of Visual Odometry with Integrated GPS . . 86
4.4.3 Integration of GPS Carrier Phase Observations 89
4.5 Conclusion 93

5 Quality of Dense Stereo with Fisheye Cameras 95
5.1 Introduction 95
5.2 Related Work 96
5.3 Two Popular Dense Stereo Methods for Perspective Cameras 98
5.4 Dense Fisheye Stereo and its Accuracy 99
5.4.1 Fisheye Model 99

5.4.2 Epipolar Rectification 100
5.4.3 3D Point Cloud with Uncertainty 101
5.5 Improved Stochastic Observation Model 103
5.5.1 Variance Analysis 103
5.5.2 Orthogonality Improvement 105
5.6 Experimental Evaluation 105
5.6.1 Variance Analysis 106
5.6.2 Orthogonality of Planes 108
5.6.3 Application Examples 108
5.6.4 Remarks 109
5.7 Conclusion 110

6 Discussion 111
6.1 Conclusion 111
6.2 Future Work 112
6.2.1 Integration of Inequality Constraints for Far Points 112
6.2.2 Modeling of Unstable Multi-camera Systems 113
6.2.3 Deep Learning Approaches 113

List of Figures 115

Bibliography 119

1 Introduction

In recent years much progress has been seen in the development of small low weight multirotor unmanned aerial vehicles (UAVs), which has led to a broad variety of systems with a simple mechanical assembly at reasonable costs. Multirotor UAVs allow to record highly overlapping images from almost terrestrial camera positions to oblique and nadir aerial images due to the ability to navigate slowly, hover, and rapidly change the direction of movement to capture images at nearly any possible position and direction. Multirotor copters thus are bridging the gap between terrestrial and traditional aerial image acquisition and are therefore ideally suited to enable easy and safe data collection, even in the close vicinity of inaccessible objects and in complex or hazardous environments. Typical fields of application reach from agriculture and environmental monitoring, surveying tasks for mining, archeology or architecture as well as inspection and assessment of objects that are difficult and dangerous to reach for human operators.

Autonomously operating multirotors demand a fast localization for navigation to actively explore unknown areas and to create maps. The design of an online pose estimation system for lightweight multirotors is challenging for the following reasons: First, the sensors and computers have to be lightweight and are often not comparable to high-quality sensors and powerful computers used on ground robots. Second, the motion characteristics of multirotors lead to full six degrees of freedom and several simplifying assumptions that are reasonable for wheeled robots cannot be made. Third, autonomous multirotors require good pose estimates at high frequencies and in near real-time to allow for a stable control of the platform. In addition to that, we are – at least for surveying applications – interested in building a model that accurately reflects the real geometry of the sensed environment.

For many outdoor localization applications the high precision of differential GPS sensors in combination with high frequency IMU is used. However, the drawbacks of GPS sensors include the sensitivity to blackouts due to gaps in GPS signal coverage and hard to model multi-path effects, such that GPS sensors may go down in performance or even entirely on various occasions. In such situations, localization needs to rely on IMU measurements, but highly accurate and long-term stable INS sensors are still quite heavy and therefore not usable in all kinds of application as on lightweight UAVs. This and the high cost of precise INS motivate the use of inexpensive vision based sensors for localization.

The ability to observe a large area in front of a camera is important for several applications. As a result of that, monocular and stereo cameras with a large field of view are becoming more and more popular. Examples include surveillance systems, humanoid robots (Bennewitz et al., 2006, Kita, 2011b, Maier et al., 2013) and unmanned aerial vehicles, see Figure 1.3. Camera systems with a large field of view mainly use wide-angle or fisheye lenses, mirrors, multiple cameras or rotating cameras. Fisheye lenses are an attractive choice as they offer several advantages in the image acquisition process. They record a large field of view at each time of exposure, they avoid difficult to calibrate mirrors, they are comparably robust from a mechanical point of view and are available at small form factors.

Visual odometry or visual SLAM (simultaneous localization and mapping) systems are supposed to work in real-time on an ordered sequence of images, e.g. acquired on a mobile robot to obtain the robot's pose, see Taketomi et al. (2017) for a review on recent developments. Bundle adjustment is a central part of most visual odometry or visual SLAM systems as it yields more efficient results in terms of accuracy and computational cost compared to other filtering techniques when employing keyframes, see Strasdat et al. (2012). Such systems are often divided into a front and back end. The front end comprises algorithms and data structures to detect and match image features, and to obtain approximate values for camera and feature parameters. The back end typically employs bundle adjustment to refine all parameters to obtain a statistically optimal solution exploiting all observations.

1.1 Motivation

The methods described in this thesis have been developed within the project Mapping on Demand. The project Mapping on Demand[1] aims at the investigation, development and testing of methods for the fast and autonomous semantic mapping of objects in an area defined by a user inquiry exploiting the advantages of a lightweight multirotor which allows the mapping of inaccessible objects (Klingbeil et al., 2014).

Maps are a central tool for making informed decisions in many applications. Multirotors equipped with high resolution cameras are already used for surveying applications, like precision farming (Xiang and Tian, 2011), infrastructure inspection (Merz and Kendoul, 2011) or archaeological site recording (Eisenbeiss et al., 2005), to obtain models that sufficiently reflect the real geometry. However, the aerial vehicle needs to be operated by a human to avoid collisions and the maps are not delivered on demand, e.g. in situations when they are needed immediately.

The required technology to solve these tasks can be subsumed under the notion Mapping on Demand. Mapping on Demand includes all the autonomously running processes, methods and tools to obtain new sensor data, to use them in combination with existing data, and to map spatial phenomena into a model in a sufficient amount of time. This procedure relieves the user of having to navigate the multirotor to explore the environment while supplying the user with a visualization of the object to derive specific conclusions

---

[1] The project Mapping on Demand has been founded by the German Research Foundation (DFG) for six years under research unit FOR 1505 and started in January 2012.

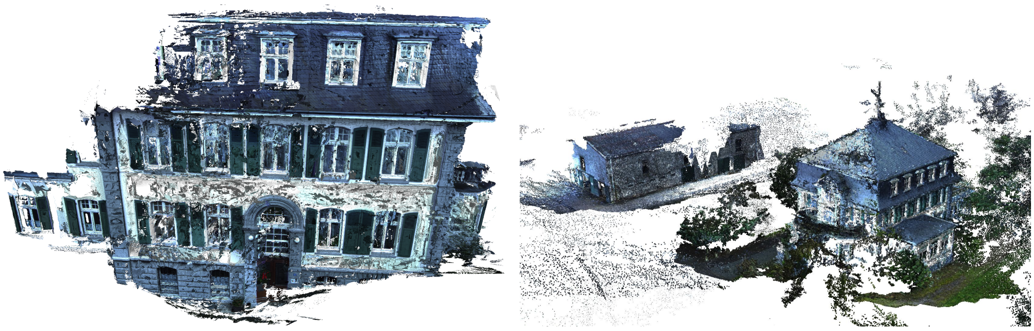


Figure 1.1: Dense 3D surface reconstruction from images acquired by an autonomously flying UAV in the project Mapping on Demand.

in a timely manner.

The deployed methods in this project aim at making application-specific models available in time, including the uncertainty of the derived data. For autonomous navigation, obstacle detection, exploration and the semantic mapping of three-dimensional objects we make use of GPS carrier phase measurements, laser information and the visual information acquired by cameras.

The actual dense mapping task runs on a ground station to relieve the onboard PC. The ground station is connected via WiFi to fetch the images of the high-resolution camera with its on-board determined georeferenced pose. This allows a simultaneously running fast incremental bundle adjustment on the ground station to determine an accurate and georeferenced pose for each image. After bundle adjustment, a dense surface reconstruction on the basis of the images and poses is executed in near-real time. A reconstructed 3D surface is depicted in Figure 1.1.

In order to achieve an autonomous operation of an UAV to acquire images for surface reconstruction, several real-time systems must interoperate. The navigation system steers the vehicle into positions, where images need to be taken for the mapping task. These positions are specified by the exploration module on the basis of the already built map to make accurate photogrammetric reconstruction possible. How to get into these positions is specified by the path planning module, which in turn avoids obstacles sensed by the obstacle perception module that uses laser information and objects seen in the camera images.

These modules rely on an accurate and reliable pose estimate. Especially the copter control and navigation demand a fast determination of the current pose of the multirotor, which therefore needs to be determined on-board of the UAV. This thesis proposes, describes and evaluates a system for fast and effective pose estimation and mapping for UAVs employing GPS, IMU and camera information.

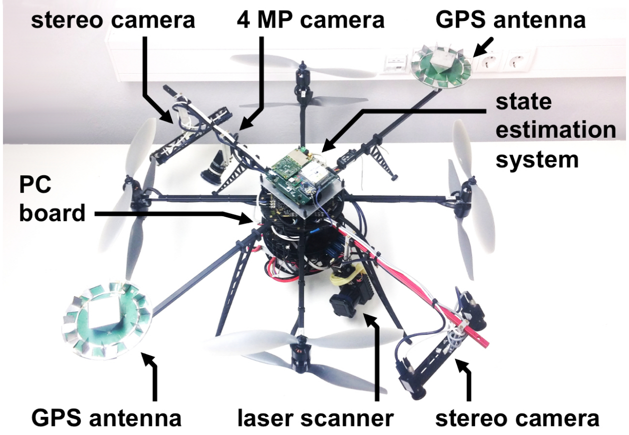


Figure 1.2: Illustration of the multirotor and its sensor setup. One stereo pair is looking forward and one backwards providing a wide field of view.

1.2 The Copter Design

This section illustrates the design decisions made in the context of the project Mapping on Demand regarding the sensor choice and configuration, especially for the visual odometry task for pose estimation.

The on-board sensing of a lightweight multirotor has generally to be designed with regards to its limitations in size and weight. The vehicle is based on a MK OktoXL platform from HiSystems with a maximum total weight of 5 kg. We use a coaxial rotor setup, in which each of four arms has two motors, one facing up and one facing down, running in opposite directions. This provides sufficient stability and power and allows us to mount sensors onto the remaining four arms.

The multirotor carries a dual frequency GPS board, an IMU, a compass, two stereo camera pairs with fisheye lenses, a rotating 3D laser scanner, a real-time processing unit and a compact PC for on-board computations, see Figure 1.2. A four megapixel camera provides images for the actual mapping task, where the environment is reconstructed in three dimensions from images, using highly accurate bundle adjustment as the basis for a subsequent dense surface estimation, computed on the ground station. We do not use the four megapixel camera for visual odometry.

Using pairs of fisheye cameras allows to capture a large field of view stereoscopically. The two stereo cameras on the copter are used besides the laser scanner for obstacle perception and besides the GPS-unit and IMU for fast pose estimation. Obstacle detection is essential for autonomous navigation of the copter. This thesis, however, focuses on the utilization of the sensors used for fast pose estimation. The four cameras with Lensagon BF2M15520 fisheye lenses, each having a field angle of up to 185°, capture four image sequences with a frame rate of 10 Hz in a synchronized way. The basis between the cameras amounts to 20 cm, providing highly overlapping views at each time of exposure, see Figure 1.3, allowing to determine depth information of objects for obstacle avoidance.

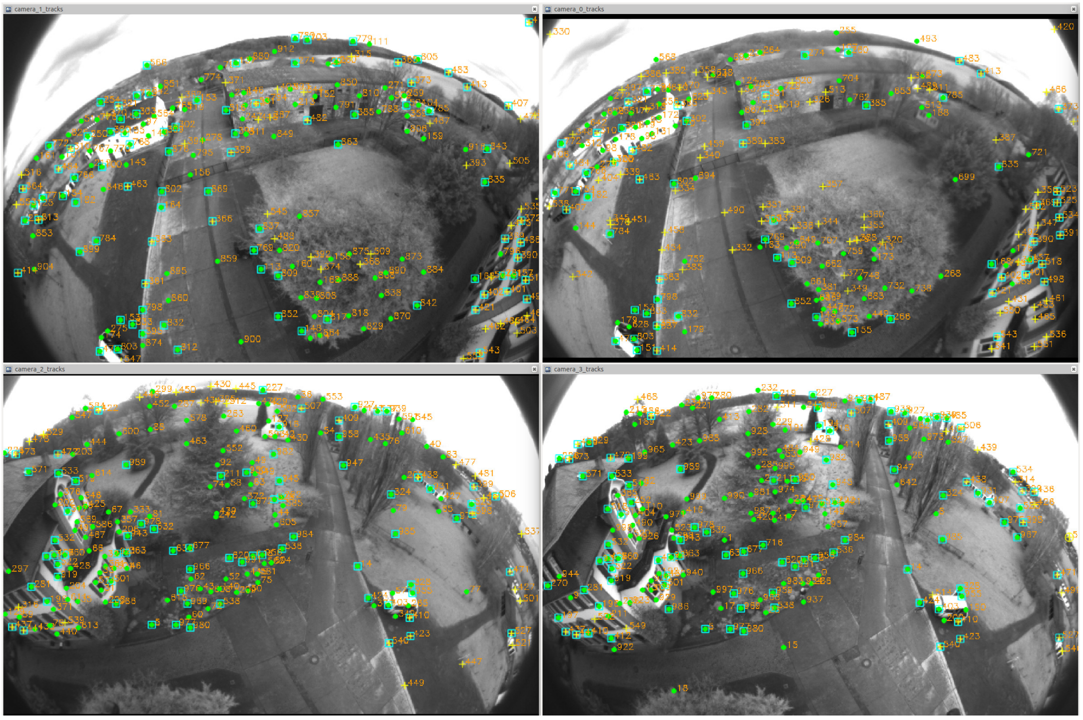


Figure 1.3: A synchronized triggered frame set of the UAV's four fisheye cameras. Each image contains around 200 feature points that are tracked using a KLT tracker.

The multi-camera system consisting of two stereo pairs, one looking ahead and one looking backwards, has, besides the advantage for obstacle detection, several advantages for pose estimation. The large field of view allows to observe scene points over long periods of time since these cameras are less prone to lose tracking, and provides better intersection geometry.

The monochromatic images of each camera are captured at 10 Hz with a resolution of 1280×1024 pixels. The limited on-board processing power requires highly efficient algorithms for image processing to obtain precise and robust estimates of the camera poses and a sparse reconstruction of the environment as fast as possible.

1.3 Contribution and Organization

1.3.1 Contributions

The contribution of this thesis consists of the following three innovative components.

The first contribution is a novel approach to bundle adjustment which addresses three issues that the classical approach is not capable of:

- We use an extended version of the projective collinearity equations which allows us to exploit multi-view camera systems by constraining the mutual orientations between the cameras. This model allows us to calibrate the mutual orientations with a rigorous bundle adjustment.

- We use ray directions as observations instead of image points. This way, we do not need to rely on a specific projection model, which allows us to process bundles of rays acquired with any central projection camera, for example omnidirectional fisheye cameras.
- We perform parameter estimation in the tangent space of spherically normalized homogeneous coordinates, which enables us to optimize unknown scene points at infinity, e.g. at the horizon, in one rigorous bundle adjustment. Such points have the great potential to stabilize the estimation of the camera's orientation. Other approaches relying on Euclidean coordinates are prone to numerical issues leading to instabilities or singularities.

Secondly, we employ our approach to bundle adjustment in a highly integrated system for incremental pose estimation and mapping on light-weight UAVs. Our system is able to effectively incorporate camera information with GPS carrier phase measurements and inertial sensor readings in our real-time SLAM system on the UAV on the level of raw observations. In contrast to existing systems, we fuse the image data with measured GPS carrier phase ranges, which allows us to even exploit measurements in underconstrained situations, i.e. if only two or three satellites are visible and standard GPS receivers report a GPS loss and cannot estimate a solution. The estimation is done in a statistically sound manner and provides accurate 6 DoF pose estimates of the platform as well as accurate 3D locations of the feature points.

Thirdly, we present an approach for re-using existing methods for dense stereo matching with fisheye cameras, which has the great advantage that highly optimized existing methods can be applied as a black-box without modifications even with cameras that have field of view of more than 180°. We provide a detailed accuracy analysis of the obtained dense stereo results, which requires a realistic stochastic model for the disparities of the matched image points. Core of the contribution therefore is a rigorous variance component estimation to optimally estimate the variance of the disparity at a point as a function of the distance of that image point to the image center. This way, we are able to use an improved stochastic model to compute the accuracy of the 3D points.

1.3.2 Organization

The thesis is organized in six chapters. The related work is given in the three chapters Chap. 3, Chap. 4 and Chap. 5.

In Chap. 2, we introduce technical aspects relevant for this thesis. We begin with uncertain projective geometry, proceed with aspects of image geometry, least-squares estimation and follow up with incremental estimation.

In Chap. 3, we work out a more general approach to bundle adjustment which allows to include points at infinity and to process image data of multi-camera systems consisting of omnidirectional cameras, e.g. with fisheye lenses. Our approach to bundle adjustment allows to estimate the system calibration of multi-camera systems consisting of multiple cameras with different projection centers.

In Chap. 4, we introduce our visual odometry system for incremental pose estimation and sparse mapping on a light-weight UAV with an omnidirectional multi-camera system. The visual odometry makes use of an incremental version of the bundle adjustment modeled in Chap. 3 which operates on keyframes. We optionally integrate GPS carrier phase information even in underconstrained situations, e.g. if only two satellites are visible. Fast pose estimation on frame rate is realized by robust resection on the incrementally refined map of 3D point coordinates.

The methods described in the preceding chapters yield a sparse reconstruction. Chap. 5 evaluates dense reconstruction with a fisheye stereo camera. We derive a measure for the uncertainty of the image point observations in fisheye images in relation to the viewing direction using variance component estimation.

Finally, Chap. 6 concludes this thesis with a discussion and proposes future work in the thesis' field of research.

1.4 Publications

Parts of this thesis have been published in journal articles as well as in conference and workshop proceedings. The content of the following chapters is partly taken from the listed publications.

Chapter 3:

[1] Schneider, J., Schindler, F., and Förstner, W. (2011). Bündelausgleichung für Multikamerasysteme. In *Proc. of the Annual Conf. of the German Society for Photogrammetry, Remote Sensing and Geoinformation (DGPF)*, pages 119–127

[2] Schneider, J., Schindler, F., Läbe, T., and Förstner, W. (2012). Bundle Adjustment for Multi-camera Systems with Points at Infinity. In *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume I-3, pages 75–80

[3] Schneider, J. and Förstner, W. (2013). Bundle Adjustment and System Calibration with Points at Infinity for Omnidirectional Camera Systems. *Photogrammetrie – Fernerkundung – Geoinformation (PFG)*, 2013(4):309–321

[4] Schneider, J., Stachniss, C., and Förstner, W. (2017). On the Quality and Efficiency of Approximate Solutions to Bundle Adjustment with Epipolar and Trifocal Constraints. In *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume IV-2/W3, pages 81–88

Chapter 4:

[5] Schneider, J., Läbe, T., and Förstner, W. (2013). Incremental Real-time Bundle Adjustment for Multi-camera Systems with Points at Infinity. In *ISPRS Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XL-1/W2, pages 355–360

[6] Nieuwenhuisen, M., Dröschel, D., Schneider, J., Holz, D., Läbe, T., and Behnke, S. (2013). Multimodal Obstacle Detection and Collision Avoidance for Micro Aerial Vehicles. In *Proc. of the European Conf. on Mobile Robotics (ECMR)*, pages 7–12

[7] Schneider, J. and Förstner, W. (2014). Real-Time Accurate Geo-Localization of a MAV with Omnidirectional Visual Odometry and GPS. In *Computer Vision - ECCV 2014 Workshops*, volume 8925 of *Lecture Notes in Computer Science (LNCS)*, pages 271–282

[8] Klingbeil, L., Nieuwenhuisen, M., Schneider, J., Eling, C., Dröschel, D., Holz, D., Läbe, T., Förstner, W., Behnke, S., and Kuhlmann, H. (2014). Towards Autonomous Navigation of an UAV-based Mobile Mapping System. In *Proc. of the Int. Conf. on Machine Control and Guidance (MCG)*, pages 136–147

[9] Schneider, J., Läbe, T., and Förstner, W. (2014). Real-Time Bundle Adjustment with an Omnidirectional Multi-Camera System and GPS. In *Proc. of the Int. Conf. on Machine Control and Guidance (MCG)*, pages 98–103

[10] Schneider, J., Eling, C., Klingbeil, L., Kuhlmann, H., Förstner, W., and Stachniss, C. (2016a). Fast and Effective Online Pose Estimation and Mapping for UAVs. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, pages 4784–4791

Chapter 5:

[11] Schneider, J., Stachniss, C., and Förstner, W. (2016c). On the Accuracy of Dense Fisheye Stereo. *IEEE Robotics and Automation Letters (RA-L)*, 1(1):227–234

[12] Schneider, J., Stachniss, C., and Förstner, W. (2016b). Dichtes Stereo mit Fisheye-Kameras. In *UAV 2016 – Vermessung mit unbemannten Flugsystemen*, volume 82 of *Schriftenreihe des DVW*, pages 247–264. Wißner Verlag

[13] Beekmans, C., Schneider, J., Läbe, T., Lennefer, M., Stachniss, C., and Simmer, C. (2016). Cloud Photogrammetry with Dense Stereo for Fisheye Cameras. *Atmospheric Chemistry and Physics (ACP)*, 16(22):14231–14248

1.5 Notation

The notation of this thesis follows the notation used by Förstner and Wrobel (2016).

To distinguish between geometric entities and their mathematical representation, we write geometric entities with calligraphic letters, e.g. points as $\mathcal{X}$ or transformations as $\mathcal{M}$. To distinguish between Euclidean coordinates and homogeneous coordinates of point $\mathcal{X}$, we use a bold and italic letter for Euclidean coordinate vectors, e.g. $\boldsymbol{x}$, and bold and upright letters for homogeneous coordinate vectors, e.g. $\mathbf{x}$. Transformations are denoted with capital letters without serifs: we use italic letters for transformations which can be applied to Euclidean coordinates, e.g. rotation matrix $\textsf{\textit{R}}$, and upright letters for homogeneous transformations, as the projection matrix $\textsf{P}$. In some passages, we underline stochastic variables to make them explicit, e.g. a stochastic rotation matrix is denoted as $\underline{\textsf{\textit{R}}}$.

The table on the next page provides an overview for future reference.

Symbol	Meaning
Vectors	
$\boldsymbol{x}$	nonhomogeneous vector
$\mathbf{x}$	homogeneous vectors
$\mathbf{x}, \mathbf{y}, \mathbf{z}$	homogeneous 3-vectors of points in 2D
$\mathbf{X}, \mathbf{Y}, \mathbf{Z}$	homogeneous 4-vectors of points in 3D
$\boldsymbol{x}_0, \boldsymbol{x}_h$	Euclidean, homogeneous part of the homogeneous coordinate vector $\mathbf{x}$
$\boldsymbol{X}_0, \boldsymbol{X}_h$	Euclidean, homogeneous part of the homogeneous coordinate vector $\mathbf{X}$
Matrices	
I_n	$n \times n$ unit matrix
J_r	Jacobian $\partial \mathbf{x} / \partial \boldsymbol{x}_r$, with reduced vector $\boldsymbol{x}_r$ of $\mathbf{x}$
J_s	Jacobian $\partial \mathbf{x}^s / \partial \mathbf{x}$ of spherical normalization
R	3×3 rotation matrix or $U \times U$ upper triangular matrix
K	3×3 homogeneous calibration matrix
M	4×4 homogeneous motion matrix
P	3×4 homogeneous projection matrix
Estimation	
$\boldsymbol{l}$	N-vector of observations in an estimation procedure
$\boldsymbol{x}$	parameter vector in an estimation procedure with U unknowns
A	$N \times U$ design matrix, Jacobian w.r.t. parameters
N	$U \times U$ normal equation matrix
$\mathsf{\Sigma}_{xy}$	$N \times N$ covariance matrix of stochastic variables $\underline{\boldsymbol{x}}$ and $\underline{\boldsymbol{y}}$
$(\cdot)^a$	vector or matrix of initial values within iterative estimation procedure
$\widehat{(\cdot)}$	estimated value
$\tilde{(\cdot)}$	true value
Operators	
$\mathrm{Diag}(\cdot)$	diagonal matrix of vector
$\mathsf{N}(\cdot)$	operator for spherical normalization of vectors
$\mathsf{N}^e(\cdot)$	operator for Euclidean normalization of homogeneous vectors
$\mathrm{null}(\cdot), \mathrm{null}^{\mathsf{T}}(\cdot)$	orthonormal matrix: basis vectors of null space as columns, transposed
$\mathsf{S}(\boldsymbol{a})$	3×3 skew symmetric matrix depending on 3-vector $\boldsymbol{a}$
$(\cdot)^{\mathsf{T}}$	transpose
$(\cdot)^{-\mathsf{T}}$	transposed of inverse matrix

2 Basic Techniques

In its first section, this chapter introduces aspects of uncertain projective geometry used in this thesis. Euclidean representations of uncertain geometric entities appear as observations and are commonly used because of their intuitive nature, uncertain homogeneous entities are key to simplify geometric operations and to handle elements at infinity.

The second section introduces basic geometry issues of the interior orientation of central projection cameras, especially of central projection cameras with fisheye lenses. Fisheye cameras provide a large viewing angle at each single shot, which cannot be achieved with conventional lenses. We will employ fisheye cameras in this thesis both for pose estimation and scene reconstruction.

The third section recaps weighted least-square estimation, which leads to best unbiased estimators for the parameters of a model, given Gaussian distributed observations. Numerical methods allow to solve the least-square estimation very efficiently, but need to iteratively relinearize in the case of a nonlinear functional model.

In the fourth section, we show how to incrementally solve the weighted least-squares problem by efficiently incorporating new observations when they arrive. We will introduce the iSAM2 algorithm (incremental Smoothing and Mapping) by Kaess et al. (2012) which allows for incremental least-squares estimation and, in the nonlinear case, performs relinearization only where needed.

2.1 Uncertainty of Homogeneous Representations and Transformations

In this section, we introduce important aspects of projective geometry which we will employ in this thesis. Using homogeneous representations in projective space provides several advantages, for example for geometric construction, transformation, estimation and variance propagation. The transition from Euclidean space to projective space is always possible, but the other way around is not generally possible as the projective space includes entities at infinity. We will formulate the uncertainty of the geometric entities with second order statistics, thus in case of a Gaussian distribution we can use a covariance matrix to describe the uncertainty. For a broad and more detailed introduction, please refer to Förstner and Wrobel (2016, Chap. 5/6).

In this thesis, homogeneous coordinates will provide us the following central advantages:

- points at infinity can be numerically represented, which is not possible with Euclidean coordinates,

- line preserving transformations can be written as matrix vector products, which offer a simplified concatenation and inversion,
- linearization and error propagation of vectors and matrices are easy as most geometric operations with homogeneous coordinates have bilinear forms.

2.1.1 Uncertain Homogeneous Points

A point $\mathcal{X}$ can be represented with a Euclidean or homogeneous coordinate vector. Uncertain Euclidean representations appear at the beginning, e.g. as sensor readings, or at the end of a processing chain. Uncertain homogeneous entities are key for simplifying geometric reasoning and for handling elements at infinity. Given the coordinates of a point $\boldsymbol{x}$ with Euclidean parameterization with its covariance matrix Σ_{xx}, we can immediately derive the uncertain homogeneous coordinates by

$$\mathbf{x} = \begin{bmatrix} \boldsymbol{x} \\ 1 \end{bmatrix}, \quad \Sigma_{\mathbf{xx}} = \begin{bmatrix} \Sigma_{xx} & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix}. \tag{2.1}$$

Note that the last element of $\mathbf{x}$ is not stochastic, the covariance matrix $\Sigma_{\mathbf{xx}}$ is therefore rank deficient.

Homogeneous coordinates are by definition invariant with respect to a multiplication with scalar $\lambda \neq 0$. Therefore $\mathbf{x}$ and $\lambda\mathbf{x}$ represent the same point:

$$\mathbf{x} = \lambda \begin{bmatrix} \boldsymbol{x} \\ 1 \end{bmatrix} = \begin{bmatrix} \boldsymbol{x}_0 \\ x_h \end{bmatrix}. \tag{2.2}$$

Here $\boldsymbol{x}_0$ specifies the Euclidean part and scalar x_h the homogeneous part of a homogeneous point vector. For several applications, as estimation, one wants to restrict the freedom of scaling. The ambiguity of scaling can be reduced by normalizing. *Euclidean normalization* leads to a vector with last element $x_h = 1$, which makes the unique Euclidean properties in $\boldsymbol{x}_0$ explicit. *Spherical normalization* leads to a unit vector with length 1 and – in contrast to Euclidean normalization – still allows to represent points at infinity with $x_h = 0$.

Spherical normalization, denoted with index s, of a homogeneous point vector $\mathbf{x}$ reads as

$$\mathbf{x}^s = \mathsf{N}(\mathbf{x}) = \frac{1}{|\mathbf{x}|}\mathbf{x} \tag{2.3}$$

such that $|\mathbf{x}^s| = 1$. Note that the negative vector $-\mathbf{x}^s$ represents the same point but the homogeneous point vector points to the opposite direction. In our application we need to take care of the sign, for example when points need to be located in front of a camera. Following the concept of oriented projective geometry, the sign of the scaling factor must be positive for two points to be identical, i.e. $\lambda > 0$.

The transition of the covariance matrix of an arbitrarily scaled vector to that of a spherically normalized vector reads as

$$\Sigma_{\mathrm{x}^s\mathrm{x}^s} = J_s \Sigma_{\mathrm{xx}} J_s^{\mathsf{T}} \qquad \text{with} \qquad J_s(\mathbf{x}) = \frac{\partial \mathsf{N}(\mathbf{x})}{\partial \mathbf{x}} = \frac{1}{|\mathbf{x}|}\left(I - \frac{\mathbf{x}\mathbf{x}^{\mathsf{T}}}{|\mathbf{x}|^2}\right). \tag{2.4}$$

Spherical normalization leads to a homogeneous coordinate vector on the unit sphere which has the property of a closed manifold. Förstner (2012) exploits this property to represent the uncertainty of homogeneous points with full rank in the tangent space, which leads to so called reduced coordinates. Reduced coordinates allow the testing and estimation in projective space with a minimal set of parameters and is free of singularities. We will use this minimal representation in Sec. 3.6 to overcome the rank deficiency of the covariance matrix in Eq. (2.4) for parameter estimation. In the following, we will assume homogeneous point vectors $\mathbf{x}$ to be spherically normalized and neglect – if unambiguous– the index s.

2.1.2 Uncertain Rotations

Rotations are a central operation in geometry with various mathematical representations. In this thesis, we will use the elements of a rotation matrix to represent rotations.This section briefly outlines our convention to represent uncertain rotation matrices.

A Euclidean point $\boldsymbol{x}$ in 3D can be transformed with a 3×3 rotation matrix R on the same 3D space by

$$\boldsymbol{x}' = R\boldsymbol{x} \qquad \text{with} \qquad R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}. \tag{2.5}$$

The elements r_{ij} are constrained by orthonormal relationships, thus R is an orthogonal linear transformation, which results in $RR^{\mathsf{T}} = I$, as we have $R^{\mathsf{T}} = R^{-1}$.

An uncertain rotation matrix $\underline{R}$ could be represented by the 9-vector $\mathrm{vec}(\underline{R})$ with its 9×9 covariance matrix. As a rotation matrix has only three degrees of freedom, this covariance matrix will have rank three. To avoid overrepresentation, we will use a multiplicative partitioning of an uncertain rotation $\underline{R}$ into a mean rotation $\mathbb{E}(\underline{R})$ and a small rotation $R(\underline{\Delta \boldsymbol{r}})$, which depends on the minimal set of three small random parameters $\underline{\Delta \boldsymbol{r}}$:

$$\underline{R} = R(\underline{\Delta \boldsymbol{r}})\mathbb{E}(\underline{R}). \tag{2.6}$$

We assume the three random parameters $\underline{\Delta \boldsymbol{r}}$ to be small, thus the small rotation of $R(\underline{\Delta \boldsymbol{r}})$ is close to the identity matrix and – up to a first-order approximation – can be rewritten as $R(\underline{\Delta \boldsymbol{r}}) \approx (I_3 + S(\underline{\Delta \boldsymbol{r}}))$. This way we have

$$\underline{R} \approx (I_3 + S(\underline{\Delta \boldsymbol{r}}))\, \mathbb{E}(\underline{R}) \qquad \text{with} \qquad S(\underline{\Delta \boldsymbol{r}}) = \begin{bmatrix} 0 & -\underline{\Delta r}_3 & \underline{\Delta r}_2 \\ \underline{\Delta r}_3 & 0 & -\underline{\Delta r}_1 \\ -\underline{\Delta r}_2 & \underline{\Delta r}_1 & 0 \end{bmatrix}, \tag{2.7}$$

where the uncertainty of $\underline{R}$ can be described by the regular 3×3 covariance matrix $\Sigma_{\Delta r \Delta r}$ of the minimal set of three parameters $\underline{\Delta r}_i$ including the correlations between the elements.

In Eq. (2.6), the vector $\underline{\Delta \boldsymbol{r}}$ and its covariance matrix refer to the coordinate system rotated by $\mathbb{E}(\underline{R})$. Note that the multiplicative partitioning would allow an alternative definition with the small uncertain rotation $\underline{\Delta \boldsymbol{r}}$ on the right side of $\mathbb{E}(\underline{R})$, which leads to $\underline{R} = \mathbb{E}(\underline{R}) R(\underline{\Delta \boldsymbol{r}})$. In doing so, $\underline{\Delta \boldsymbol{r}}$ and its covariance matrix would refer to the coordinate system before $\mathbb{E}(\underline{R})$ has been applied. We stick with the definition given in Eq. (2.6).

When estimating rotations, e.g. in a bundle adjustment, we usually start with some approximate rotation R^a. The rotation matrix will be corrected by $\widehat{R} = R(\widehat{\Delta \boldsymbol{r}}) R^a$, where the small rotation $R(\widehat{\Delta \boldsymbol{r}})$ depends on a small rotation vector $\widehat{\Delta \boldsymbol{r}}$ that is to be estimated. For linearization we will again make use of first order approximation $R(\widehat{\Delta \boldsymbol{r}}) \approx I_3 + S(\widehat{\Delta \boldsymbol{r}})$. To obtain an orthonormal rotation matrix $R(\widehat{\Delta \boldsymbol{r}})$ for the multiplicative correction applied to R^a, we make use of the Cayley representation $\boldsymbol{u} = \frac{1}{2}\widehat{\Delta \boldsymbol{r}}$ and obtain with

$$R(\widehat{\Delta \boldsymbol{r}}) = (I_3 + S(\boldsymbol{u}))\,(I_3 - S(\boldsymbol{u}))^{-1} \tag{2.8}$$

a valid rotation matrix.

Finally, note that the three entries of a small rotation vector $\Delta \boldsymbol{r}$ can be interpreted as the Euler angles around the x-, y- and z-axis of the rotated system, thus the covariance matrix $\Sigma_{\Delta r \Delta r}$ provides a direct interpretation of the uncertainty of the rotation.

2.1.3 Uncertain Homogeneous Spatial Motions

A spatial motion consists of a rotation R and translation $\boldsymbol{t}$, and may be seen as a rigid body transformation. In Euclidean 3D space the motion of a point $\boldsymbol{x}$ to $\boldsymbol{x}'$ is applied by a mixture of multiplication and addition

$$\boldsymbol{x}' = R\boldsymbol{x} + \boldsymbol{t}. \tag{2.9}$$

Using homogeneous coordinates $\mathbf{x}$ we can perform this displacement with a bilinear matrix-vector product

$$\mathbf{x}' = \mathsf{M}\mathbf{x} \qquad \text{with} \qquad \mathsf{M} = \begin{bmatrix} R & \boldsymbol{t} \\ \mathbf{0}_3^\mathsf{T} & 1 \end{bmatrix} \tag{2.10}$$

with the homogeneous motion matrix M.

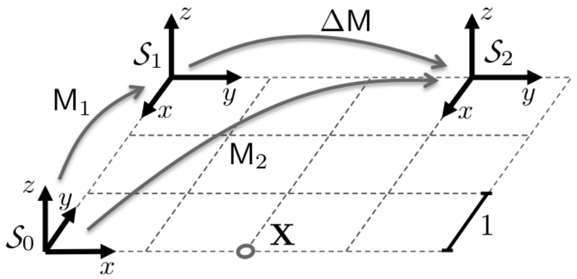


Figure 2.1: Illustration of the coordinate systems $\mathcal{S}_1$ and $\mathcal{S}_2$ which are displayed by M_1 and M_2 with respect to $\mathcal{S}_0$, respectively. In the corresponding example motion M_2 can be obtained by concatenating M_1 either with $\Delta\mathsf{M}_L$ on its left or $\Delta\mathsf{M}_R$ on its right.

Concatenation The homogeneous bilinear representation allows to compute the concatenation of several consecutive motions as matrix-matrix products. The concatenation of two motion matrices may be realized by left or right multiplication, which leads in general to different results as matrix multiplication is not commutative. The following example illustrates the geometric effect of the concatenation by left or right multiplication.

Consider two motion matrices M_1 and M_2, both of them contain a rotation of $-90°$ around z-axis and a translation of 3 in y-direction. Motion matrix M_2 additionally contains a translation of 3 in x-direction. Figure 2.1 visualizes coordinate system $\mathcal{S}_1$ displaced by M_1 and coordinate system $\mathcal{S}_2$ displaced by M_2, both with respect to the origin $\mathcal{S}_0$.

Motion matrix M_2 can be obtained by concatenating M_1 with a new motion matrix $\Delta\mathsf{M}_L$ on its left or with $\Delta\mathsf{M}_R$ on its right:

$$\mathsf{M}_2 = \Delta\mathsf{M}_L\mathsf{M}_1 \qquad \text{or} \qquad \mathsf{M}_2 = \mathsf{M}_1\Delta\mathsf{M}_R. \tag{2.11}$$

If M_1 is *multiplied on its left* by $\Delta\mathsf{M}_L$, motion matrix $\Delta\mathsf{M}_L$ describes the displacement in coordinate frame $\mathcal{S}_0$, thus $\Delta\mathsf{M}_L$ is a pure translation of 3 in x-direction.

If M_1 is *multiplied on its right* by $\Delta\mathsf{M}_R$, motion matrix $\Delta\mathsf{M}_R$ describes the displacement in coordinate frame $\mathcal{S}_1$, thus $\Delta\mathsf{M}_R$ is a pure translation of 3 in y-direction.

As the example illustrates, the knowledge of the coordinate frame a motion refers to is essential for the interpretation of its covariance information, e.g. of estimated motion parameters, or to integrate uncertain observations of different sensors into a joint estimation problem.

Coordinate Transformation A motion matrix M describes the translation and rotation of one coordinate system into another one, as in the example motion M_2 transforms $\mathcal{S}_0$ into $\mathcal{S}_2$. We can employ the inverse motion matrix M^{-1} to transform homogeneous coordinates into another coordinate system: a point $\mathcal{X}$ given in $\mathcal{S}_0$, e.g. with homogeneous coordinates

$\mathbf{X} = [2, 0, 0, 1]^\mathsf{T}$, see Figure 2.1, is transformed into coordinate system $\mathcal{S}_2$ by

$$\mathbf{X}' = \mathsf{M}_2^{-1}\mathbf{X} = \begin{bmatrix} 3 \\ -1 \\ 0 \\ 1 \end{bmatrix} \qquad \text{with} \qquad \mathsf{M}^{-1} = \begin{bmatrix} R^\mathsf{T} & -R^\mathsf{T}\boldsymbol{t} \\ \mathbf{0}_3^\mathsf{T} & 1 \end{bmatrix}. \tag{2.12}$$

Note that M_2^{-1} can be regarded as a motion which displaces $\mathcal{S}_2$ into $\mathcal{S}_0$.

Uncertain Motions The number of elements of the homogeneous motion matrix M in Eq. (2.10) amounts to 16, but a spatial motion has only six degrees of freedom. To represent the uncertainty of an uncertain motion matrix $\underline{\mathsf{M}}$ with the minimal set of six parameters, we use the multiplicative partitioning

$$\underline{\mathsf{M}} = \mathsf{M}(\underline{\Delta\boldsymbol{r}}, \underline{\Delta\boldsymbol{t}})\mathbb{E}(\underline{\mathsf{M}}) = \mathsf{T}(\underline{\Delta\boldsymbol{t}})\mathsf{R}(\underline{\Delta\boldsymbol{r}})\mathbb{E}(\underline{\mathsf{M}}) \tag{2.13}$$

with a small rotation

$$\mathsf{R}(\underline{\Delta\boldsymbol{r}}) = \begin{bmatrix} R(\underline{\Delta\boldsymbol{r}}) & \mathbf{0}_3 \\ \mathbf{0}_3^\mathsf{T} & 1 \end{bmatrix} \tag{2.14}$$

and small translation

$$\mathsf{T}(\underline{\Delta\boldsymbol{t}}) = \begin{bmatrix} I_3 & \underline{\Delta\boldsymbol{t}} \\ \mathbf{0}_3^\mathsf{T} & 1 \end{bmatrix}, \tag{2.15}$$

which depend on the small random parameters $\underline{\Delta\boldsymbol{r}}$ and $\underline{\Delta\boldsymbol{t}}$, and with mean motion

$$\mathbb{E}(\underline{\mathsf{M}}) = \begin{bmatrix} \mathbb{E}(\underline{R}) & \mathbb{E}(\underline{\boldsymbol{t}}) \\ \mathbf{0}_3^\mathsf{T} & 1 \end{bmatrix} \tag{2.16}$$

consisting of mean rotation $\mathbb{E}(\underline{R})$ and mean translation $\mathbb{E}(\underline{\boldsymbol{t}})$.

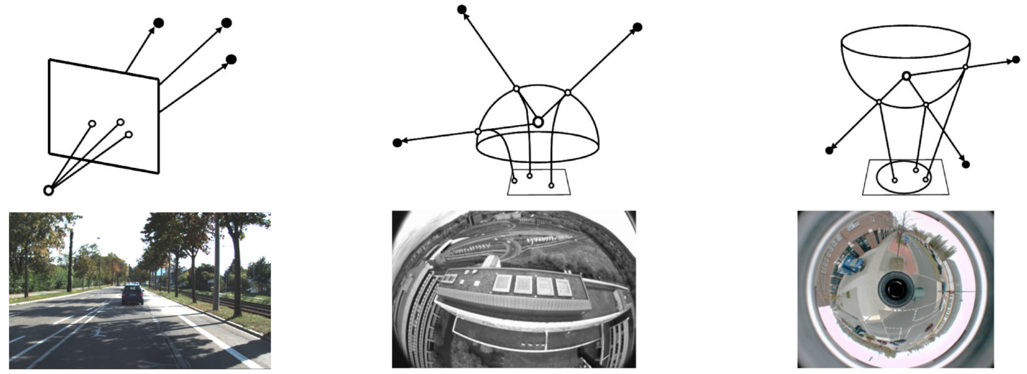
The multiplicative concatenation in Eq. (2.13) leads to

$$\underline{\mathsf{M}} = \mathsf{M}(\underline{\Delta\boldsymbol{r}}, \underline{\Delta\boldsymbol{t}})\mathbb{E}(\mathsf{M}) = \begin{bmatrix} R(\underline{\Delta\boldsymbol{r}})\mathbb{E}(R) & R(\underline{\Delta\boldsymbol{r}})\mathbb{E}(\boldsymbol{t}) + \underline{\Delta\boldsymbol{t}} \\ \mathbf{0}_3^\mathsf{T} & 1 \end{bmatrix}. \tag{2.17}$$

This way, the random rotation parameter vector $\underline{\Delta\boldsymbol{r}}$ and its covariance matrix $\Sigma_{\Delta r \Delta r}$ are oriented in the reference coordinate system as $\mathsf{R}(\underline{\Delta\boldsymbol{r}})$ is concatenated by left multiplication. The random translation vector $\underline{\Delta\boldsymbol{t}}$ and its covariance matrix $\Sigma_{\Delta t \Delta t}$ are oriented in the coordinate system rotated with small correction $\mathsf{R}(\underline{\Delta\boldsymbol{r}})$ as $\mathsf{T}(\underline{\Delta\boldsymbol{t}})$ is concatenated again by left multiplication.

2.2 Basic Image Geometry

Digital cameras project the 3D world onto a 2D image plane by sensing the intensity of light traveling along rays on a CCD array. During the time of exposure, each CCD sensor



(a) perspective camera[2] (b) fisheye camera (c) catadioptric camera[3]

Figure 2.2: Illustration of different mapping principles of central projection cameras with a single projection center: (a) perspective camera, (b) fisheye camera and (c) catadioptric camera with hyperbolic mirror.

element receives a certain number of photons that induce a certain current, which in turn can be transformed into a discrete intensity value. The intensity value and its position on the CCD sensor represent a pixel, the smallest picture element of the sensed raster image.

In this thesis, we use central projection cameras which have a single view point – the so called projection center – at which all sensed ray intersect. Thus, each sensed pixel measures the irradiance of light passing through the projection center in one particular direction. The camera specific projection of light onto the image plane is described by the interior orientation of the camera. In case the interior orientation of the camera is known, that is when the camera is calibrated, one can retrieve for each sensed 2D image point $\boldsymbol{x}'$ the projected 3D ray direction ${}^{c}\mathbf{x}$ in the camera coordinate system.

The camera specific light projection depends on the used optical system. There are numbers of real world cameras with different constructions and projection principles (Förstner and Wrobel, 2016, Chap. 11), which can be advantageous in certain applications. Figure 2.2 shows three examples of cameras with optical systems that follow different projection principles: a perspective camera, a fisheye camera and a catadioptric camera using a hyperbolic mirror. Perspective cameras are commonly used because of their natural mapping following a pinhole model which preserves straight lines similar to the visual perception of humans. Additionally, an ideal perspective mapping can be easily modeled using the intercept theorem. Fisheye and catadioptric cameras allow to map a wide field of view, which can be advantageous for visual odometry especially in the close vicinity of objects (Zhang et al., 2016). Fisheye lenses are an attractive choice as they avoid the difficulty to calibrate mirrors which can cause caustics, they are comparably robust from a mechanical point of view, can be mounted on standard CCD- or CMOS-cameras without high technical effort and are available in very small and lightweight form factors.

---

[2] Karlsruhe Institute of Technology, Kitti Dataset, digital image, accessed on 14 July 2018, `http://www.cvlibs.net/datasets/kitti/raw_data.php`

[3] Humboldt Universität Berlin, Institut für Informatik, digital image, accessed on 14 July 2018, `https://www2.informatik.hu-berlin.de/~wwwcv/website/img/projects/alternative1.jpg`

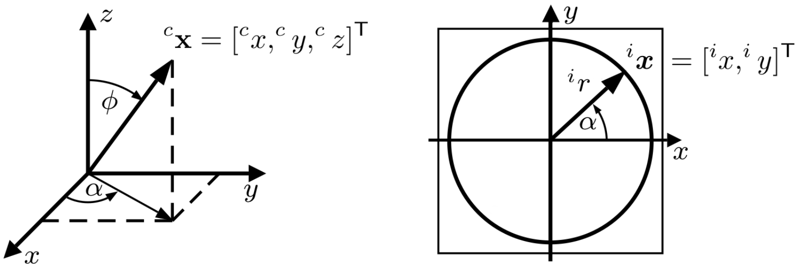

z
${}^c\mathbf{x} = [{}^cx, {}^cy, {}^cz]^\mathsf{T}$
ϕ
α
y
x
y
${}^i\boldsymbol{x} = [{}^ix, {}^iy]^\mathsf{T}$
${}^ir$
α
x

Figure 2.3: Relation between a camera ray direction and a projected image point. Left: Camera ray ${}^c\mathbf{x}$ specified by angles ϕ and α in camera frame with optical axis z. Right: Relation between direction angles and image point coordinates ${}^i\boldsymbol{x}$ with radial distance ${}^ir$.

2.2.1 Interior Orientation

To describe the interior orientation of a camera, we follow Abraham and Förstner (2005) and separate the transformation of a 3D camera ray into image coordinates into a projection model and a distortion model. The *projection model* describes the ideal and error free transformation of 3D camera ray ${}^c\mathbf{x}$ into 2D image coordinates ${}^i\boldsymbol{x}$, and should be chosen according to the projection properties of the lens to prevent too large distortions. Projection models in general are radial symmetric in relation to the optical axis. The center of best symmetry is therefore the intersection of the optical axis and the image plane, which is the origin of the coordinate system of the conditioned image. Let the direction of the 3D camera ray ${}^c\mathbf{x} = [{}^cx, {}^cy, {}^cz]^\mathsf{T}$ be specified by two angles ϕ and α, as depicted in Figure 2.3. The radial distance ${}^ir$ of an projected conditioned image point ${}^i\boldsymbol{x} = \left[{}^ix, {}^iy\right]^\mathsf{T}$ to the origin then only depends on the angle ϕ between the 3D camera ray and the optical axis

$$
{}^i\boldsymbol{x} = \begin{bmatrix} {}^ix \\ {}^iy \end{bmatrix} = {}^ir(\phi) \begin{bmatrix} \cos(\alpha) \\ \sin(\alpha) \end{bmatrix} . \tag{2.18}
$$

Classical cameras can be well approximates with the perspective projection model whereas fisheye lenses are designed to follow a equi-distance projection model. As we will use both kind of cameras, we will employ both projection models in this thesis. Additionally, in Chap. 5 we will employ the stereographic projection model, which allows to rectify an image pair of a stereo camera with fisheye lenses, such that standard dense stereo methods can be applied.

All three projection models basically differ in the definition of the radial projection function ${}^ir(\phi)$. The latter two can be used to model the projection of fisheye and catadioptric cameras with a large field of view.

The classical *perspective projection model* follows the projection of a pinhole camera, where the radial projection function ${}^ir(\phi) = \tan(\phi)$ increases with the tangent of the incident angle ϕ. The projection ${}^c\mathbf{x} \mapsto {}^i\boldsymbol{x}$ and the inverse transformation ${}^i\boldsymbol{x} \mapsto {}^c\mathbf{x}$ then read as

$$ {}^{i}\boldsymbol{x} = \frac{1}{{}^{c}z} \begin{bmatrix} {}^{c}x \\ {}^{c}y \end{bmatrix} \qquad \text{and} \qquad {}^{c}\mathbf{x} = \begin{bmatrix} {}^{i}\boldsymbol{x} \\ 1 \end{bmatrix} . \tag{2.19} $$

Note that the perspective projection only allows a limited range of ϕ as for large angles the tangent rapidly grows towards infinity. Perspective projection is beneficial for several image processing applications as it preserves straight lines, i.e. straight lines in the 3D scene are projected as straight lines in the 2D sensor.

The *equi-distance projection model* projects each ray into an image point with a radial distance ${}^{i}r(\phi) = \theta$ to the distortion center. The distance is proportional to the incident angle ϕ, which allows the projection of a wide field of view with incident angles larger than 90°. The projection ${}^{c}\mathbf{x} \mapsto {}^{i}\boldsymbol{x}$ and the inverse transformation ${}^{i}\boldsymbol{x} \mapsto {}^{c}\mathbf{x}$ then read as

$$ {}^{i}\boldsymbol{x} = \frac{\operatorname{atan2}\left({}^{c}r, {}^{c}z\right)}{{}^{c}r} \begin{bmatrix} {}^{c}x \\ {}^{c}y \end{bmatrix} \qquad \text{and} \qquad {}^{c}\mathbf{x} = \begin{bmatrix} \dfrac{\sin({}^{i}r)}{{}^{i}r} {}^{i}\boldsymbol{x} \\ \cos({}^{i}r) \end{bmatrix} . \tag{2.20} $$

Note that ${}^{i}r(\phi) = \theta$ is a monotonously increasing function, which allows to project rays with incident angles even larger than 90°. Most fisheye lenses are designed to project the incident angles in a proportional distance to the center of symmetry, which makes the equi-distance projection model a good approximation for fisheye cameras.

Thirdly, we introduce the *stereographic projection model*. Its radial projection function ${}^{i}r = \tan(\phi/2)$ increases only with the tangent of the incident angle $\phi/2$, thus allows, as the equi-distance model, to project rays with incident angles larger than 90°. However, the radial distance increases vastly for angles $\phi > 90°$. The projection ${}^{c}\mathbf{x} \mapsto {}^{i}\boldsymbol{x}$ and the inverse transformation ${}^{i}\boldsymbol{x} \mapsto {}^{c}\mathbf{x}$ read as

$$ {}^{i}\boldsymbol{x} = \frac{1}{|{}^{c}\mathbf{x}|^2 + {}^{c}z} \begin{bmatrix} {}^{c}x \\ {}^{c}y \end{bmatrix} \qquad \text{and} \qquad {}^{c}\mathbf{x} = \frac{1}{1 + {}^{i}r^2} \begin{bmatrix} 2\,{}^{i}\boldsymbol{x} \\ 1 - {}^{i}r^2 \end{bmatrix} . \tag{2.21} $$

Stereographic projection is conformal, which means the projection preserves angles, i.e. angles between intersecting lines in the scene have the same angle in the projected image.

Up to now, we have related the ideal projection to the center of symmetry in the coordinate system of conditioned image points, which is the *principal point* $\boldsymbol{h} = [h_x, h_y]^{\mathsf{T}}$ in the pixel coordinate system of the actually acquired image. The principal point needs to be estimated in a camera calibration procedure together with the *principal distance* c, which in case of a distortion-free projection is the distance of the projection center to the image plane. A camera whose interior orientation can be described only with parameters $\boldsymbol{h}$ and c is called a Euclidean camera as the geometric elements in the image plane follow Euclidean geometry.

The distortions induced by imperfections of the camera lens need to be modeled with a *distortion model*, which can be set up e.g. with polynomials describing radial symmetric, asymmetric or tangential distortions, or with tangential polynomials (Abraham and Förstner, 2005). Imperfections of the sensor alignment are usually parametrized by an additional scale difference and a shear of the axis. We combine the latter two deviations from a Euclidean camera into the location dependent corrections $\Delta\boldsymbol{x}(\boldsymbol{x}) = [\Delta x(x,y),\, \Delta y(x,y)]^{\mathsf{T}}$. A general way to represent the deviations is by using a distortion lookup-table. This means, for every measured pixel coordinate $\boldsymbol{x}$ in the actually acquired image there exists a distortion vector to obtain the image point coordinates $\boldsymbol{x}'$ of a Euclidean camera. The distortion lookup-table can be obtained in a camera calibration process, e.g. as proposed by Abraham and Hau (1997). A measured image point $\boldsymbol{x}$ can be corrected by the non-linear deviations from a Euclidean camera by

$$\boldsymbol{x}' = \boldsymbol{x} + \Delta\boldsymbol{x}(\boldsymbol{x}) \; . \tag{2.22}$$

Typically, for non-integer coordinates the distortions are interpolated.

Given the pixel coordinates of image point $\boldsymbol{x}'$, the relation to conditioned image point ${}^{i}\boldsymbol{x}$ is given by

$${}^{i}\boldsymbol{x} = \frac{1}{c}(\boldsymbol{x}' - \boldsymbol{h}) \; , \tag{2.23}$$

where $\boldsymbol{x}'$ is shifted by the principal point $\boldsymbol{h}$ into the origin of the image coordinate system, and scaled with the principal distance c. Note that the principal distance c scales the radial projection function to the actual pixel coordinates, and therefore defines the maximum measurable angle ϕ in the image.

2.2.2 Camera Calibration

The photogrammetry community developed a large number of camera calibration techniques to recover the intrinsic parameters. Many analytical camera calibration techniques have been proposed in the 60s and 70s, which brought standard techniques, primarily with metric cameras in mind. With the increased use of non-metric cameras, especially the computer vision community has turned the attention to developments aiming at efficient, autonomous, versatile and accurate camera calibration techniques for non-metric cameras (Fraser, 2001).

A common and often cited approach for camera calibration has been proposed by Tsai (1987), which served as a base model for several modified and extended approaches, for example by Zhang (2000). Based on the method of Zhang (2000), the very popular test-field-based Camera Calibration Toolbox[4] has been developed by Jean-Yves Bouguet, which is also included as a C implementation in the OpenCV library (Bradski and Kaehler, 2008). The method requires images of a planar chessboard rig taken from different perspectives and applies a self-calibrating bundle adjustment.

In general, test-field-based calibration procedures detect image points and identify them in images taken from different views. In this work, we use the calibration technique developed by Abraham and Hau (1997), which detects ellipses in the images taken of a test-field with circular retro-reflective markers, see Figure 2.4. To calibrate cameras with a large field of view, e.g., fisheye cameras, we use the test-field with three orthogonal planes. In order to calibrate cameras which follow a perspective projection, we use the planar test-field with a stamp which provides some 3D structure. The method includes the estimation of the intrinsic parameters, namely the principal distance, principal point, scale difference and a shear of the image axis as well as parameters of the distortion model, e.g. the radial and tangential distortion coefficients proposed by Brown (1971). A more detailed description of the calibration method is given in Abraham (1999).

---

[4] `http://www.vision.caltech.edu/bouguetj/calib_doc/`

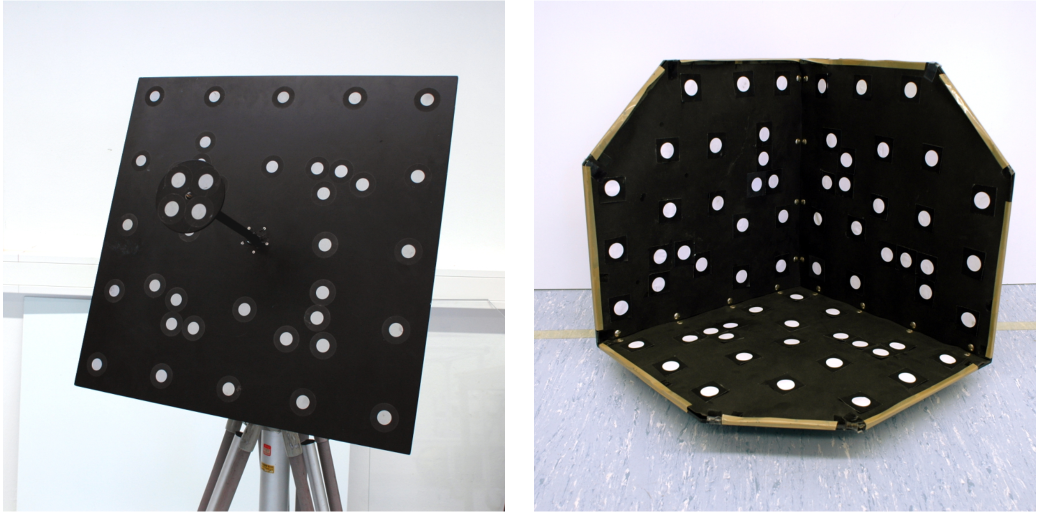


Figure 2.4: The two test-fields with retro-reflective circular markers for camera calibration. The planar test-field with a stamp (left) is used for the calibration of perspective cameras, the cubic test-field for the calibration of fisheye cameras.

Scaramuzza et al. (2006) propose another development to recover the intrinsic parameters of omnidirectional cameras as catadioptric and fisheye cameras and provide the OCamCalib Toolbox[5] for Matlab. The calibration technique employs – as Bouguet's toolbox – a planar chessboard, but it does not allow to set up a distortion model to take lens distortions into account. A comprehensive overview of more camera models and calibration methods is given by Puig et al. (2012).

In practice, the intrinsic parameters are estimated often once as camera specific constants. The influence of thermal changes and other external influences on the intrinsic parameters is then considered as negligibly small. However, a regular calibration is advisable as long term studies and experiments show a drift of the intrinsic camera parameters. Hence, online calibration techniques become more and more important. Hemayed (2003) provide a good survey on camera self-calibration techniques.

---

[5] `https://sites.google.com/site/scarabotix/ocamcalib-toolbox`

2.3 Weighted Least-squares Estimation

There are different techniques to describe the estimation of unknown parameters from given observations, one among these is the well known least-squares estimation. Least-squares estimation is a very effective numerical method and leads to best unbiased estimators for linear relationship and observations disturbed by Gaussian noise.

In this section, we will briefly introduce the Gauss–Markov model, which contains a functional and stochastic model to frame the observation process. The functional model specifies the assumed relation between the acquired observations and the unknown parameters as an explicit function, which usually results from physical or geometrical laws. The stochastic model specifies the statistical properties of the observation process, and is assumed to be sufficiently described by the first and second moments of a normal distribution. The Gauss–Markov model covers many practical estimators including maximum likelihood (ML) and maximum a posteriori (MAP) estimators. For a detailed introduction into estimation theory with emphasis on least squares estimation please refer to the books of Koch (1999, Chap. 3) or Förstner and Wrobel (2016, Chap. 4).

2.3.1 Estimation with Non-linear Gauss–Markov Model

The Gauss–Markov model starts from N observations $\boldsymbol{l} = [l_n]$, $n = 1, ..., N$, which are assumed to be a sample of a multivariate Gaussian distribution $\mathcal{N}(\tilde{\boldsymbol{l}}, \Sigma_{ll})$ around a true but unknown observation vector $\tilde{\boldsymbol{l}}$ with a symmetric and positive definite covariance matrix Σ_{ll}. Due to the noise induced by the observation process, there are in general no parameters $\boldsymbol{x}$ for which a functional model $\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{l}$ holds. Therefore the goal is to find corrections $\widehat{\boldsymbol{v}}$ for observations $\boldsymbol{l}$ and best estimates $\widehat{\boldsymbol{x}}$ such that the relation

$$\boldsymbol{f}(\widehat{\boldsymbol{x}}) = \boldsymbol{l} + \widehat{\boldsymbol{v}} = \widehat{\boldsymbol{l}} \tag{2.24}$$

between the fitted observations $\widehat{\boldsymbol{l}} = \boldsymbol{l} + \widehat{\boldsymbol{v}}$ and the estimated parameters $\widehat{\boldsymbol{x}}$ holds and the weighted sum of the squared residuals

$$\Omega(\widehat{\boldsymbol{x}}) = \widehat{\boldsymbol{v}}^{\mathsf{T}} \Sigma_{ll}^{-1} \widehat{\boldsymbol{v}} \tag{2.25}$$

is minimum.

The optimization problem therefore reads as

$$\widehat{\boldsymbol{x}} = \operatorname{argmin}_{\boldsymbol{x}} (\boldsymbol{f}(\boldsymbol{x}) - \boldsymbol{l})^{\mathsf{T}} \Sigma_{ll}^{-1} (\boldsymbol{f}(\boldsymbol{x}) - \boldsymbol{l}) \,, \tag{2.26}$$

which leads to estimated parameters $\widehat{\boldsymbol{x}}$, which have minimal variance, i.e., are best.

For a nonlinear function $\boldsymbol{f}(\boldsymbol{x})$ the solution is iterative. Starting from initial values $\widehat{\boldsymbol{x}}^{(\nu=0)}$

for the estimated parameters $\widehat{\boldsymbol{x}}$ in the first iteration $\nu = 0$ we determine updates $\widehat{\Delta\boldsymbol{x}}^{(\nu=0)}$

$$\widehat{\boldsymbol{x}}^{(\nu+1)} = \widehat{\boldsymbol{x}}^{(\nu)} + \widehat{\Delta\boldsymbol{x}}^{(\nu)}\,. \tag{2.27}$$

Each following iteration solves for the updates $\widehat{\Delta\boldsymbol{x}}^{(\nu)}$ of the linearized function

$$\boldsymbol{l} + \widehat{\boldsymbol{v}}^{(\nu)} = \boldsymbol{f}(\widehat{\boldsymbol{x}}^{(\nu)}) + \mathsf{A}\widehat{\Delta\boldsymbol{x}}^{(\nu)} \tag{2.28}$$

with Jacobian matrix

$$\mathsf{A} = \left.\frac{\partial \boldsymbol{f}(\boldsymbol{x})}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\widehat{\boldsymbol{x}}^{(\nu)}} \tag{2.29}$$

evaluated at initial parameters $\widehat{\boldsymbol{x}}^{(\nu)}$. With the reduced observations

$$\Delta\boldsymbol{l}^{(\nu)} = \boldsymbol{f}(\widehat{\boldsymbol{x}}^{(\nu)}) - \boldsymbol{l} \tag{2.30}$$

we can determine the unknown parameter updates $\widehat{\Delta\boldsymbol{x}}^{(\nu)}$ from the normal equation system

$$\mathsf{A}^{\mathsf{T}}\Sigma_{ll}^{-1}\mathsf{A}\widehat{\Delta\boldsymbol{x}}^{(\nu)} = \mathsf{A}^{\mathsf{T}}\Sigma_{ll}^{-1}\Delta\boldsymbol{l}^{(\nu)} \tag{2.31}$$

for example with Cholesky factorization (Golub and Loan, 1996, Sec. 4.2).

The corrections of the observations can be determined linearly after each iteration by

$$\widehat{\boldsymbol{v}}^{(\nu)} = \mathsf{A}\widehat{\Delta\boldsymbol{x}}^{(\nu)} - \Delta\boldsymbol{l}^{(\nu)} \tag{2.32}$$

which after convergence are equal to the non-linearly determined corrections

$$\widehat{\boldsymbol{v}} = \boldsymbol{f}(\widehat{\boldsymbol{x}}) - \boldsymbol{l}\,. \tag{2.33}$$

We arrive at $\widehat{\boldsymbol{x}} := \widehat{\boldsymbol{x}}^{(\nu)}$ in case of convergence, i.e., $\widehat{\Delta\boldsymbol{x}} \to \boldsymbol{0}$. Convergence is achieved if all updates for parameters $\widehat{\boldsymbol{x}}$ are small compared to their standard deviation, $|\Delta\widehat{x}_u/\sigma_{x_u}| < T_c$, e.g. with a threshold $T_c = 0.01$, requiring the updates to be less than 1 % of their standard deviation.

The full covariance matrix of the estimated parameters is obtained by

$$\Sigma_{\widehat{x}\widehat{x}} = \widehat{\sigma}_0^2(\mathsf{A}^{\mathsf{T}}\Sigma_{ll}^{-1}\mathsf{A})^{-1} \tag{2.34}$$

with estimated variance factor

$$\widehat{\sigma}_0^2 = \frac{\widehat{\boldsymbol{v}}^{\mathsf{T}}\Sigma_{ll}^{-1}\widehat{\boldsymbol{v}}}{R} \tag{2.35}$$

with the redundancy $R = N - U$ of the optimization problem with the number N of observations, i.e. the dimension of vector $\boldsymbol{l}$, and the number U of unknown parameters, i.e. the dimension of vector $\boldsymbol{x}$.

2.3.2 Robust Estimation

The presented least squares estimation is highly sensitive to outliers in the observations as the weighted sum of squared residuals is minimized. Observations are usually considered as outliers if the realized measurement is significantly out of the dispersion range of the expected value. Within an estimation procedure, outliers can be detected based on the magnitude of a computed residual $\widehat{v}_n$. Following Baarda (1967) for uncorrelated observations the test value

$$T_n = \frac{\widehat{v}_n}{\sigma_{\widehat{v}_n}} \tag{2.36}$$

with

$$\Sigma_{\widehat{v}\widehat{v}} = \Sigma_{ll} - A\Sigma_{\widehat{x}\widehat{x}}A^{\mathsf{T}} \tag{2.37}$$

follows the standard normal distribution $T_n \sim \mathcal{N}(0,1)$ if there are no gross errors in the observations. Assuming all observations to have an equally high influence on the parameter vector, one could use σ_{l_n} instead of $\sigma_{\widehat{v}_n}$ in Eq. (2.36). If T_n deviates significantly from the standard normal distribution, the corresponding observation can be assumed to be an outlier, thus should be eliminated from the estimation process. Rigorous testing for outliers by means of hypothesis testing is treated by Koch (1999).

Alternatively, the influence of high residuals on the cost function can be reduced by robust estimation techniques as reweighting procedures, which can be directly incorporated into the iterative estimation procedure of non-linear least-squares. Assuming again stochastically uncorrelated observations, Eq. (2.25) can be rewritten as

$$\Omega(\boldsymbol{x}) = \sum_n \frac{1}{2}\left(\frac{v_n}{\sigma_{l_n}}\right)^2 = \sum_n \rho(y_n) \tag{2.38}$$

with normalized residuals $y_n = v_n/\sigma_{l_n}$ and piecewise influence functions

$$\rho(y_n) = \frac{1}{2}y_n^2\,. \tag{2.39}$$

To arrive at a robust estimation procedure, Huber (1981) proposes using a probability density function for the observations which consists of a normal distribution in the middle and of a Laplace distribution at the ends. This way the density function has more probability mass at the ends and thus allows to model a certain amount of gross errors in the observations. The modified influence function $\rho_H(y_n)$ is defined as

$$\rho_H(y_n) = \begin{cases} \frac{1}{2}y_n^2 & \text{for } |y_n| \le k, \\ k(|y_n| - \frac{k}{2}), & \text{otherwise,} \end{cases} \tag{2.40}$$

where k is a constant which needs to be defined according to the amount of outliers in the observations. Koch (1999) recommends using $k = 1.5$ for 4 % outliers, and $k = 2$ for less than 1 % outliers.

The target function in Eq. (2.38) does not need to be changed when we reweight the variances $\sigma^2_{l_n}$ with a weighting function $w(y_n)$ after each iteration step ν by

$$\sigma^{2,(\nu+1)}_{l_n} = w(y_n^{(\nu)})\sigma^{2,(\nu)}_{l_n} \tag{2.41}$$

with

$$w(y) = \frac{\partial\rho(y)/\partial y}{y} = \begin{cases} 1 & \text{for } |y_n| \leq k \\ \frac{k}{|y_n|} & \text{otherwise} \end{cases} \tag{2.42}$$

according to the properties of M-estimators examined by Hampel et al. (1986, Chap. 2).

2.4 Incremental Estimation

For applications which need to run online, as online SLAM, the least-squares estimation needs to be executed in real-time, thus the process of incorporating new sensor observations and solving for parameters needs to be computationally efficient. In this section, we will briefly review the incremental smoothing and mapping approach iSAM by Kaess et al. (2012) for fast incremental parameter estimation. The approach allows to integrate new observations and parameters into the optimization without the need to rebuild the entire normal equation system.

To solve the least squares problem in Eq. (2.31), usually a Cholesky factorization of the normal equation matrix is applied, which allows to efficiently obtain the solution by forward and back substitution, see Golub and Loan (1996, Sec. 3.1). The QR factorization is an alternative approach and is directly applied to the Jacobian matrix A in Eq. (2.29), as shown in the following section. We can easily extend the QR factorization incrementally with new observations and parameters by employing Givens rotations, which will be topic of the second section. Subsequently, we will briefly introduce the iSAM2 algorithm, which allows for incremental least squares estimation with incremental reordering and incremental relinearization on a reduced set of affected variables to retain sparseness and full accuracy.

2.4.1 QR Matrix Factorization

In the following, we will use the decorrelated Jacobian matrix

$$\mathsf{A} := \Sigma_{ll}^{-1/2}\mathsf{A}\,, \tag{2.43}$$

which preserves the sparsity pattern of A. The QR factorization of an $N{\times}U$ matrix A yields

$$A = Q \begin{bmatrix} R \\ 0 \end{bmatrix}, \tag{2.44}$$

where R is an upper triangular $U{\times}U$ matrix and Q is an orthogonal $N{\times}N$ matrix as $Q^\mathsf{T} Q = I$ (Golub and Loan, 1996, Sec. 5.2). The matrix R is called *square root information matrix*, as the information matrix, i.e. the normal equation matrix, is given by $N = R^\mathsf{T} R$. With the factorization of the decorrelated Jacobian of Eq. (2.43) we can rewrite the normal equation system in Eq. (2.31) as

$$A^\mathsf{T} A \widehat{\Delta\boldsymbol{x}} = A^\mathsf{T} \Delta\boldsymbol{l} \tag{2.45}$$

$$\begin{bmatrix} R^\mathsf{T} & 0^\mathsf{T} \end{bmatrix} Q^\mathsf{T} Q \begin{bmatrix} R \\ 0 \end{bmatrix} \widehat{\Delta\boldsymbol{x}} = \begin{bmatrix} R^\mathsf{T} & 0^\mathsf{T} \end{bmatrix} Q^\mathsf{T} \Delta\boldsymbol{l} \tag{2.46}$$

$$\begin{bmatrix} R^\mathsf{T} & 0^\mathsf{T} \end{bmatrix} \begin{bmatrix} R \\ 0 \end{bmatrix} \widehat{\Delta\boldsymbol{x}} = \begin{bmatrix} R^\mathsf{T} & 0^\mathsf{T} \end{bmatrix} Q^\mathsf{T} \Delta\boldsymbol{l} \tag{2.47}$$

$$\begin{bmatrix} R \\ 0 \end{bmatrix} \widehat{\Delta\boldsymbol{x}} = Q^\mathsf{T} \Delta\boldsymbol{l}\,. \tag{2.48}$$

What we see here is that QR factorization simplifies the least squares problem to the linear system

$$R \widehat{\Delta\boldsymbol{x}} = \boldsymbol{d} \qquad \text{with} \qquad \begin{bmatrix} \boldsymbol{d} \\ \boldsymbol{e} \end{bmatrix} := Q^\mathsf{T} \Delta\boldsymbol{l} \tag{2.49}$$

with U-dimensional vector $\boldsymbol{d}$. We can solve for $\widehat{\Delta\boldsymbol{x}}$ with simple back-substitution as R is upper triangular. The computationally expensive part is done by QR decomposition.

2.4.2 Incremental Factorization with New Observations and Unknowns

When new measurements arrive, which possibly involves the estimation of additional parameters, it is more efficient to modify the previous factorization instead of updating and refactoring the Jacobian A again. The previously calculated components of R and the right hand side $\boldsymbol{d}$ of Eq. (2.49) can be reused in a subsequent stage k yielding the new system

$$\begin{bmatrix} R \,|\, 0 \\ A_k \end{bmatrix} \begin{bmatrix} \widehat{\Delta\boldsymbol{x}} \\ \widehat{\Delta\boldsymbol{x}}_k \end{bmatrix} = \begin{bmatrix} \boldsymbol{d} \\ \Delta\boldsymbol{l}_k \end{bmatrix}, \tag{2.50}$$

in which vector $\widehat{\Delta\boldsymbol{x}}_k$ contains updates for U_k new parameters, $\Delta\boldsymbol{l}_k$ is the right hand side of N_k new observation equations with Jacobian A_k containing the decorrelated coefficients w.r.t. all parameters.

The extended system of stage k in Eq. (2.50) is not yet in the correct factorized form. A standard approach to obtain the QR factorization uses a sequence of Givens rotations to transform a general matrix into upper triangular form, as shown by Golub and Loan (1996, Sec. 5.1). With a set of orthogonal Givens rotations $\mathsf{G} = \mathsf{G}_i \mathsf{G}_{i-1} ... \mathsf{G}_1$ on Eq. (2.50) we can eliminate all non-zero entries below the main diagonal.

Typically A_k is sparse and new measurements refer only to recently added variables, such that only a few Givens rotations are required and only the right-most part of the new Jacobian A_k is populated, which leads to minor fill-in. Applying the set of Givens rotations G to Eq. (2.50) leads to

$$\mathsf{G}\begin{bmatrix} \mathsf{R} \,|\, \mathit{0} \\ \mathsf{A}_k \end{bmatrix}\begin{bmatrix} \widehat{\Delta \boldsymbol{x}} \\ \widehat{\Delta \boldsymbol{x}_k} \end{bmatrix} = \mathsf{G}\begin{bmatrix} \boldsymbol{d} \\ \Delta \boldsymbol{l}_k \end{bmatrix} \tag{2.51}$$

$$\begin{bmatrix} \mathsf{R}_k \\ \mathit{0} \end{bmatrix}\begin{bmatrix} \widehat{\Delta \boldsymbol{x}} \\ \widehat{\Delta \boldsymbol{x}_k} \end{bmatrix} = \begin{bmatrix} \boldsymbol{d}_k \\ \boldsymbol{e}_k \end{bmatrix}, \tag{2.52}$$

where R_k is a $(U+U_k)\times(U+U_k)$ upper triangular matrix and $\boldsymbol{d}_k$ a $(U+U_k)$-dimensional vector. The new equation system of stage k

$$\mathsf{R}_k \begin{bmatrix} \widehat{\Delta \boldsymbol{x}} \\ \widehat{\Delta \boldsymbol{x}_k} \end{bmatrix} = \boldsymbol{d}_k \tag{2.53}$$

can be solved efficiently for new updates with back-substitution.

Grün (1985) introduced the Givens rotations-based updating process for incremental estimation in photogrammetric on-line triangulation. The same technique serves as the basis for the incremental smoothing and mapping approach (iSAM) by Kaess et al. (2008). Kaess et al. use the incremental estimation for SLAM applications and show that linear time is needed to update the square root information matrix and to perform parameter estimation with back-substitution for typical exploration tasks with landmark and odometry measurements, where the normal equation matrix is typically band-diagonal.

Applying this procedure iteratively keeps the linearization point unchanged, but relinearization may become necessary to obtain full accuracy. Especially loop closures, which occur when previously visited places are revisited, lead to fill-in in R, which makes the incremental updates time expensive. Thus the proposed iSAM algorithm performs periodic batch steps with a full refactoring of the square root information matrix R. This allows to apply variable reordering, which maintains efficiency by avoiding fill-in, and relinearization to achieve consistency and full accuracy by updating the linearization point of all variables.

2.4.3 The iSAM2 Algorithm

Periodic batch steps with full refactoring may become too time expensive for online applications. Kaess et al. (2012) evolved the iSAM algorithm into the fully incremental iSAM2 algorithm[6], which allows for incremental relinearization and incremental variable reordering. The iSAM2 algorithm allows to obtain – up to a definable threshold – full accuracy for the parameter estimation while being entirely incremental. This is achieved by converting the optimization problem into what is called a Bayes tree, which is the probabilistic graphical model equivalent to the square root information matrix, thus – from the linear algebra perspective – the result of QR factorization. The Bayes tree preserves the sparsity and allows to realize the incremental estimation by simple editing of this tree.

In the following, we will introduce factor graphs, which allow to represent the optimization problems like Eq. (2.26) as a probabilistic graphical model. Accordingly we will look at the conversion of a factor graph into a Bayes net, which – in a linear algebra perspective – corresponds to the QR factorization. Key to the iSAM2 algorithm is the conversion of the Bayes net into the Bayes tree, which allows to effectively incorporate new observations, and – in contrast to the iSAM algorithm – to incrementally relinearize the parameters and to incrementally change the ordering of the parameters to avoid fill-in.

Basically, a factor graph represents the factorization of a probability distribution function $\phi(\boldsymbol{x})$ of an optimization problem in a graphical model (Kschischang et al., 2001) and is well suited to model sparse estimation problems. The factorization reads as

$$\phi(\boldsymbol{x}) = \prod_i \phi_i(\boldsymbol{x}_i) \tag{2.54}$$

with factor nodes ϕ_i and variable parameter nodes $\boldsymbol{x}_i$. The factor graph is a bipartite graph with edges which connect each factor node $\phi_i(\boldsymbol{x}_i)$ with its parameters $\boldsymbol{x}_i$. The parameters in $\boldsymbol{x}_i$ are a subset of all parameters in $\boldsymbol{x}$.

The factor nodes have the unnormalized probability density function

$$\phi_i(\boldsymbol{x}_i) \propto \exp\left(-\frac{1}{2}\left\|\boldsymbol{f}_i(\boldsymbol{x}_i) - \boldsymbol{l}_i\right\|^2_{\Sigma_{l_i l_i}}\right) \tag{2.55}$$

when assuming Gaussian measurement noise as in Sec. 2.3.1. Again, the factor $\boldsymbol{f}_i(\boldsymbol{x}_i)$ is the observation function of observations $\boldsymbol{l}_i \in \boldsymbol{l}$ with covariance matrix $\Sigma_{l_i l_i}$. Please note that here only the observations in $\boldsymbol{l}_i$ are assumed to be correlated. As only subsets of parameters $\boldsymbol{x}_i$ are involved in factors ϕ_i, the factorization makes the sparse structure explicit, which typically appears in applications as SLAM or bundle adjustment.

To obtain the best estimates $\widehat{\boldsymbol{x}}$, we need to maximize the probability distribution $\phi(\boldsymbol{x})$, which leads to the least-squares optimization problem

$$\widehat{\boldsymbol{x}} = \operatorname{argmin}_{\boldsymbol{x}} - \log \phi(\boldsymbol{x}) \tag{2.56}$$

$$= \operatorname{argmin}_{\boldsymbol{x}} \sum_{i=1}^{I} (\boldsymbol{f}_i(\boldsymbol{x}_i) - \boldsymbol{l}_i)^{\mathsf{T}} \Sigma^{-1}_{l_i l_i} (\boldsymbol{f}_i(\boldsymbol{x}_i) - \boldsymbol{l}_i)\,. \tag{2.57}$$

---

[6] The implementation is part of the GTSAM library: `https://collab.cc.gatech.edu/borg/gtsam/`

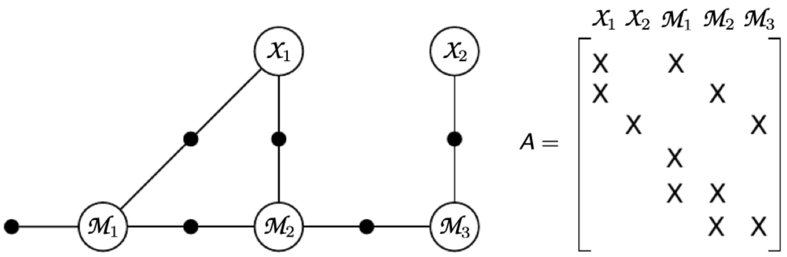
This optimization problem is equivalent to Eq. (2.26), but makes the independence relationships explicit which are encoded by the edges of a factor graph. Figure 2.5 (a) introduces an exemplary factor graph and the corresponding structure of the associated and sparsely populated Jacobian matrix A.

Kaess et al. (2012) propose to convert the factor graph of a least-squares optimization problem into a Bayes net. This is achieved by a bipartite elimination game, where variable nodes of the factor graph are eliminated sequentially and converted into variable nodes of the Bayes net. The resulting Bayes net has the same structure as the square-root information matrix R obtained by QR factorization of A if the same variable elimination ordering is applied. Figure 2.5 (b) shows the structure of R and the Bayes net after elimination of the exemplary factor graph.

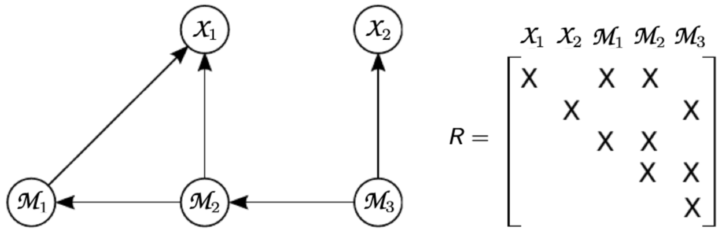
The obtained Bayes net is chordal, thus can be converted into a tree-structured graphical model. Kaess et al. (2010) propose the Bayes-tree, which is a directed clique tree and key to the incremental relinearization and reordering strategies of the iSAM2 algorithm. To obtain the Bayes tree, cliques in the chordal Bayes net are identified with the maximum cardinality search algorithm by Tarjan and Yannakakis (1984). Accordingly, the discovered cliques are converted into nodes of the Bayes tree. The resulting Bayes tree represents the elements of the square root information matrix R in a graphical model while maintaining the sparse structure as shown in Figure 2.5 (c). The root node of the Bayes tree contains the last set of variables in the elimination ordering and its clique variables. Using this data structure allows to neither form the complete matrices A nor R explicitly.

The properties of the Bayes tree are key to make the iSAM2 algorithm entirely incremental. Whenever a new measurement is added, i.e. a factor $\phi(\boldsymbol{x}_i, \boldsymbol{x}_j)$, only the paths in the Bayes tree between the cliques containing $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ as well as the root are affected. This is why only the affected part needs to be turned back into a factor graph to incorporate new factors and variables. To avoid fill-in, the new factor graph is reordered with the COLAMD algorithm by Davis et al. (2004) and accordingly a new Bayes tree is formed. The remaining unaffected sub-trees can be reattached unchanged. This way an incremental update and reordering of the Bayes tree is obtained.

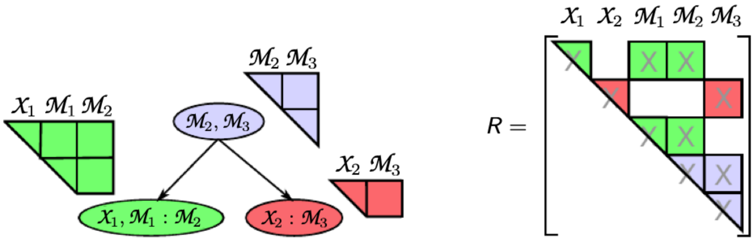
Solving for the unknown updates for the parameters works via back-substitution and starts at the root and continues to all leaves. A nearly exact but computational cost-efficient solution does not require solving for all variables, as – in applications as online SLAM or bundle adjustment – only the recently added top of the Bayes tree is affected by new factors. Processing of a sub-tree stops in case a clique is reached referring to variables whose estimated updates are smaller than a prespecified threshold β. Relinearization is performed only on the variables whose estimated update is larger than threshold β. The approach is called fluid relinearization and is done in combination with the incremental update step.


$\mathcal{X}_1$
$\mathcal{X}_2$
$\mathcal{M}_1$
$\mathcal{M}_2$
$\mathcal{M}_3$
$A =$

(a) Factor graph (left) and structure of corresponding Jacobian matrix A (right) of an exemplary optimization problem which consists of two scene points $\mathcal{X}_1$ and $\mathcal{X}_2$ which are observed from three poses $\mathcal{M}_1$, $\mathcal{M}_2$ and $\mathcal{M}_3$. Additionally, there are odometry measurements between the poses and an absolute measurement of the pose $\mathcal{M}_1$. Variable nodes are shown as large circles and factor nodes (observations) as small solid circles. The Jacobian A encodes the connections between factor and variable nodes in the corresponding rows and columns with nonzero entries.


$\mathcal{X}_1$
$\mathcal{X}_2$
$\mathcal{M}_1$
$\mathcal{M}_2$
$\mathcal{M}_3$
$R =$

(b) The Bayes net (left) results from Gaussian elimination of the factor graph with elimination order $\mathcal{X}_1, \mathcal{X}_2, \mathcal{M}_1, \mathcal{M}_2, \mathcal{M}_3$. In terms of probabilities, the Gaussian elimination converts the product of all factors in Eq. (2.55) into an equivalent product of conditional density functions $\prod_j P(\theta_j|S_j)$ of variable node θ_j given variable nodes in S_j which point in the graphical Bayes net towards θ_j. The elimination is equivalent to sparse QR factorization of A resulting in the square root information matrix R (right).


$\mathcal{M}_2$ $\mathcal{M}_3$
$\mathcal{X}_1$ $\mathcal{M}_1$ $\mathcal{M}_2$
$\mathcal{M}_2, \mathcal{M}_3$
$\mathcal{X}_2$ $\mathcal{M}_3$
$\mathcal{X}_1, \mathcal{M}_1 : \mathcal{M}_2$
$\mathcal{X}_2 : \mathcal{M}_3$
$\mathcal{X}_1$ $\mathcal{X}_2$ $\mathcal{M}_1$ $\mathcal{M}_2$ $\mathcal{M}_3$
$R =$

(c) The nodes of the Bayes tree (left) capture the clique structure of the Bayes net which depends on the chosen elimination order. Each node k defines a conditional density $P(F_k|S_k)$ with separator variables S_k containing the intersection with its parent clique and with frontal variables F_k containing the remaining variables of its clique. In terms of probabilities, the Bayes tree converts the conditional density of the Bayes net into an equivalent product $\prod_k P(F_k|S_k)$. In the nodes of the exemplary Bayes tree, frontal and separator variables are denoted as $F_k : S_k$. The sparse relationship between the conditional densities with the rows of R are indicated by color. The representation of the optimization problem in a Bayes Tree allows to efficiently exploit sparse updates. For example, updating variables in the green node only requires updating variables in the nodes towards the root, which leaves the entries marked in red unchanged.

Figure 2.5: A small optimization problem represented as a factor graph (a) which is converted into a Bayes net (b) and accordingly into a Bayes tree (c). The data structure of the Bayes tree is key to efficiently solve sparse non-linear optimization problems as bundle adjustment or SLAM. Figure and example adapted from Kaess et al. (2012).

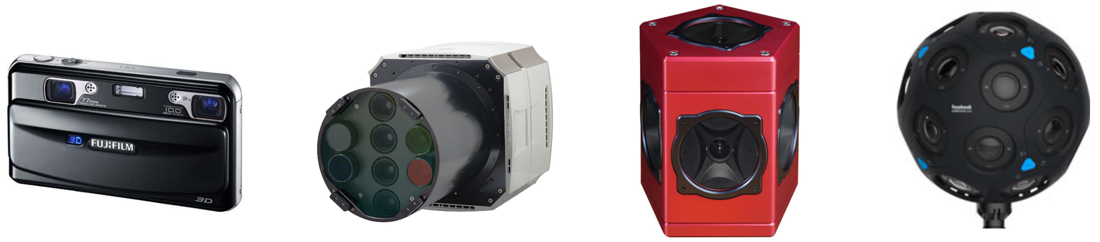
3 Bundle Adjustment for Multi-Camera Systems with Far Points

The goal of this chapter is to work out a model for bundle adjustment. The model allows to process images of omnidirectional cameras, to employ the mutual orientation of several perspective and omnidirectional cameras in a multi-camera system, and to include points which are far or even at infinity. It is based on an extended version of the projective collinearity equations, which constrain the mutual relative poses between the cameras to be fixed and makes the relative poses between the cameras explicit and thus easily allows to estimate the system calibration. The model allows to explicitly estimate the system calibration and enables an efficient maximum-likelihood estimation with points at infinity, which would cause numerical issues in the classical approach. Including observations of points at the horizon stabilizes camera orientations – especially rotations – as such points can be observed over long periods of time. Employing omnidirectional camera systems allows to cover a wide field of view, omnidirectional multi-camera systems consisting of more than one camera allow additionally to maintain a proper pixel resolution.

In the following, we will model bundle adjustment and address three issues that classical bundle adjustment approaches are not capable of:

- First, we use an extended version of the projective collinearity equations which allows to exploit multi-view camera systems consisting of several central projection cameras by constraining the mutual orientation.
- Second, we use ray directions as observations instead of image points. Instead of eliminating the scale factor of homogeneous vectors by Euclidean normalization, we will employ spherically normalized homogeneous coordinate vectors, which allows us to employ bundles of rays acquired with omnidirectional cameras.
- Third, in contrast to classical bundle adjustment approaches, we perform parameter estimation in the tangent space of spherically normalized homogeneous coordinates, which enables us to optimize unknown scene points at infinity, e.g. at the horizon, in a rigorous bundle adjustment.

This way we are able to use omnidirectional camera systems, which can be fisheye cameras and can consist of several central projection cameras with different view points. It further enables the use of image and scene points that are far away or at infinity by using a minimal representation of homogeneous coordinates. The proposed bundle adjustment is called BACS (Bundle Adjustment for Camera Systems) and has been published in



(a) Finepix Real 3D W1[8] (b) Vexcel UltraCam[9] (c) Ladybug 3[10] (d) Facebook x24[11]

Figure 3.1: Four camera designs for multi-camera systems with multiple viewpoints: (a) consumer stereo camera with two viewpoints, (b) high resolution multi-spectral camera with eight viewpoints, (c) and (d) omnidirectional multi-camera systems with six and 24 different view points.

(Schneider et al., 2012). A Matlab implementation is available.[7]

3.1 Introduction

Bundle adjustment is the work horse for orienting cameras and determining 3D points. It has a number of favorable properties: it is statistically optimal in case all statistical tools are exploited, highly efficient in case sparse matrix operations are used, useful for test field free self-calibration and can be parallelized to a high degree.

3.1.1 Multi-camera Systems

Multi-camera systems consist of several single-view cameras whose projection centers do not necessarily coincide, see Figure 3.1. This way multi-camera systems provide for example the flexibility to increase the resolution or – like omnidirectional cameras – to augment the effective aperture angle (Ladybug 3). Multi-camera systems are also used to combine cameras with different spectral sensitivities (Z/I DMC, Vexcel UltraCam) and multi-camera systems especially gain importance for the acquisition of complex 3D structures, e.g. for virtual and augmented reality applications (Facebook x24). In this thesis, we assume that the single-view cameras in a multi-camera system take images in a synchronized way and have mutually stable relative poses.

---

[7] Department of Photogrammetry, University of Bonn, accessed on 14 July 2018, `http://www.ipb.uni-bonn.de/data-software/bacs/`

[8] Fujifilm Holdings K.K., Finepix Real 3D W1, digital image, accessed on 14 July 2018, `https://www.fujifilm.eu/fileadmin/product_migration/dc/headerimage/Finepix_Real_3D_W1_51.png`

[9] Vexcel Imaging GmbH, Vexcel Ultracam, digital image, accessed on 14 July 2018, `http://www.vexcel-imaging.com/wp-content/uploads/2016/07/FalconM2_Highlight_9.jpg`

[10] Flir Systems, Inc., Ladybug 3, digital image, accessed on 14 July 2018, `http://www.vision-smart.com/uploadfile/2011/0908/20110908115820904.jpg`

[11] Jonathan Nafarrete, VRScout.com, digital image, accessed on 14 July 2018, `https://13apq3bncl82o596k2d1ydn1-wpengine.netdna-ssl.com/wp-content/uploads/2017/04/facebook-cam-x24-360.jpg`

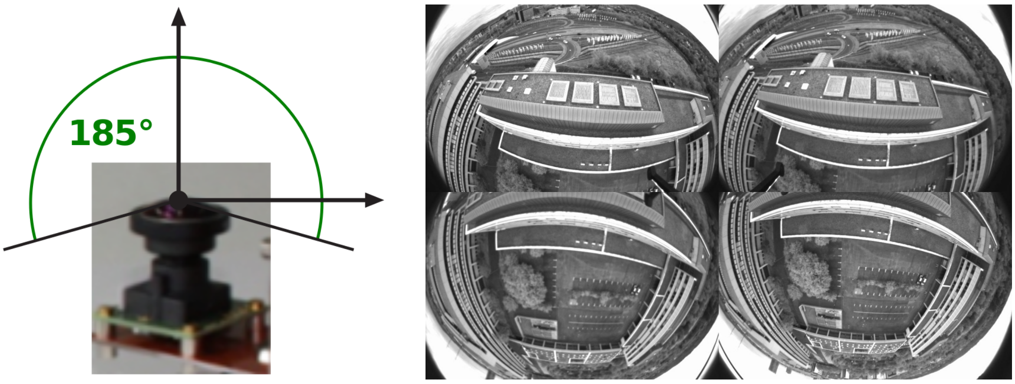


Figure 3.2: Left: A single fisheye camera is able to capture more than a half-sphere. Right: The images of four fisheye cameras in a multi-camera system with synchronized time of exposure. The multi-camera system is mounted on the multi-rotor depicted in Figure 1.2 on page 14 and provides a wide field of view.

3.1.2 Omnidirectional Cameras

Omnidirectional cameras have a viewing range of more than a half-sphere, such as omnidirectional multi-camera systems, catadioptric cameras including mirrors, and special fisheye lenses (Scaramuzza, 2008). Omnidirectional cameras can consist of several cameras, like a multi-camera system, but can also be only one single camera, such as catadioptric cameras which have a hyperbolic mirror to obtain an omnidirectional field of view. Such systems are non-central projection cameras as the projection rays do not coincide in one single point but on a common surface, the so-called caustic. Swaminathan et al. (2001) show properties of the caustic of such systems and present a way to calibrate catadioptric systems. Fisheye cameras with lenses of higher quality have nearly one single viewpoint and only a small caustic at the image border, thus it is reasonable to assume a central projection camera, i.e. to approximate the small caustic by a single viewpoint (Ying and Hu, 2004). Fisheye cameras have wide-angle lenses with a very short focal length and compared to conventional lenses the mapping does not preserve straight lines. Wide field-of-view cameras are especially beneficial for applications such as visual odometry or SLAM (Davison et al., 2004). Figure 3.2 shows the images acquired by four fisheye cameras which are assembled to a multi-camera system to cover a wide field of view.

3.1.3 Points at Infinity

Far points or points at infinity, for example points at the horizon, are effective in stabilizing the orientation of cameras, especially their rotation. Figure 3.3 shows two images of an image sequence acquired on a UAV flight, guiding the UAV along a straight street. The marked point at the horizon can be tracked over long periods of time and thus effectively stabilizes the estimation of the camera's orientation in bundle adjustment. However, the observed ray directions of such far points have small intersection angles, and due to their uncertainties the intersection may vanish at infinity, which leads to numerical issues in

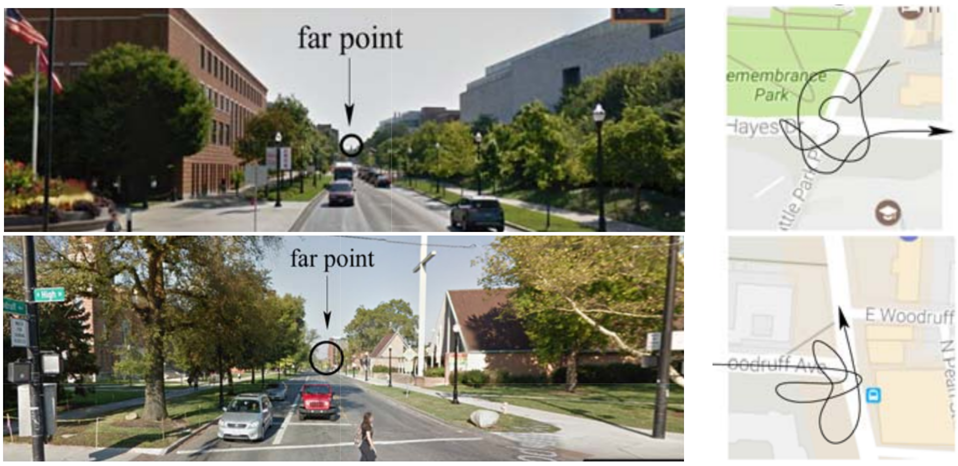


Figure 3.3: Local flight at Woodruf Ave with far point towards west end. The far point can be tracked over long periods in an image sequence and thus effectively stabilizes the orientation estimation. Figure taken from (Förstner, 2017).

the estimation process when using Euclidean coordinates. Additionally, small changes in the observed ray directions of far scene points lead to large changes in the 3D space such that the cost function becomes very flat and convergence is not guaranteed (Triggs et al., 2000, Urban et al., 2017). Thus, the classical approach to bundle adjustment requires excluding far points to avoid numerical issues and convergence difficulties. In order to exploit the power of bundle adjustment by using all available information, it therefore should be extended to allow for scene points at infinity.

Civera et al. (2008) propose the inverse depth parametrization to include scene points with small intersection angles in a standard extended Kalman filter framework for monocular SLAM, which allows an undelayed initialization. The inverse depth parametrization of a scene point $\mathcal{X}$ contains three parameters: the inverse distance $\rho = 1/d$ between $\mathcal{X}$ and its reference point $\boldsymbol{Z}$, and the two direction angles λ and θ of the spherically normalized 3D vector $\mathbf{m}(\lambda, \theta)$, $|\mathbf{m}| = 1$, which points from $\boldsymbol{Z}$ to $\mathcal{X}$ in the world coordinate system with origin $\mathcal{O}$, see Figure 3.4.

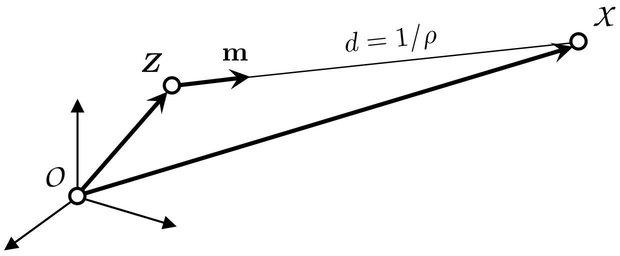


Figure 3.4: The inverse depth representation of point $\mathcal{X}$, which is possibly far or at infinity, with reference point $\boldsymbol{Z}$, spherically normalized 3D direction vector $\mathbf{m}$ and inverse distance ρ.

This way, all 3D points which are possibly far or at infinity, are represented with

$$\mathcal{X}(\boldsymbol{Z};\lambda,\theta,\rho) \ : \quad \boldsymbol{X} = \boldsymbol{Z} + \frac{1}{\rho}\,\mathbf{m}(\lambda,\theta)\ . \tag{3.1}$$

The inverse depth parametrization allows the estimation of points at infinity which have inverse distance $\rho = 0$. The choice of the coordinate system to represent a scene point depends on the maximum intersection angle between the observed ray directions. If the scene point is stable, i.e. has been observed with a large intersection angle, it can be transferred into the global coordinate system.

However, the Jacobian of $\mathbf{m}(\lambda,\theta)$ shows a singularity at the poles, which is why, possibly for each scene point $\mathcal{X}$, an adequate reference point needs to be chosen to avoid singularities in the optimization process. The function could be replaced by a direction vector which could be estimated using reduced coordinates. We use a common coordinate system for all points, which simplifies the modeling. This way, we use a parametrization which is free of any singularities and allows the estimation of points at infinity without assigning different reference points.

Another approach to bundle adjustment which allows to include observations of scene points at infinity relies on epipolar and trifocal constraints, see Schneider et al. (2017). Epipolar and trifocal constraints lead to implicit functions that enforce the intersection of bundles of rays in 3D space without explicitly representing 3D point coordinates. However, because of the implicit epipolar and trifocal constraints one needs to employ the Gauss–Helmert Model for optimization, which requires the costly determination of corrections for all observations, which is not needed in the Gauss–Markov Model, which can be employed when using the explicit collinearity equations.

3.1.4 The Idea

The classical collinearity equations for image points $\mathscr{x}'_{it}([x'_{it},\, y'_{it}]^{\mathsf{T}})$ of scene point $\mathcal{X}_i([X_i,\, Y_i,\, Z_i]^{\mathsf{T}})$ in camera t with rotation matrix $\mathcal{R}_t([r_{kk'}])$ with k and $k' = 1, ..., 3$ and projection center $\mathcal{Z}_t([X_{0t},\, Y_{0t},\, Z_{0t}])$ read as

$$x'_{it} = c\,\frac{r_{11}(X_i - X_{0t}) + r_{21}(Y_i - Y_{0t}) + r_{31}(Z_i - Z_{0t})}{r_{13}(X_i - X_{0t}) + r_{23}(Y_i - Y_{0t}) + r_{33}(Z_i - Z_{0t})} \tag{3.2}$$

$$y'_{it} = c\,\frac{r_{12}(X_i - X_{0t}) + r_{22}(Y_i - Y_{0t}) + r_{32}(Z_i - Z_{0t})}{r_{13}(X_i - X_{0t}) + r_{23}(Y_i - Y_{0t}) + r_{33}(Z_i - Z_{0t})} \tag{3.3}$$

in case of an ideal camera with principal distance c.

Obviously, these equations are not useful for far points or ideal points as small angles between rays lead to numerical instabilities or singularities. Neither are they useful for bundles of rays of omnidirectional cameras as rays perpendicular to the viewing direction, as they may occur with fisheye cameras, cannot be transformed into image coordinates. This would require different versions of the collinearity equation for different types of sensors as one would need to integrate the camera model into the bundle adjustment. Finally, the equations cannot easily be extended to systems of multiple cameras as one would need to integrate an additional motion, namely the motion from the coordinate system of the camera system to the individual camera systems.

We can avoid these disadvantages by using homogeneous coordinates $\mathbf{x}'_{it}$ and $\mathbf{X}_i$ for image and scene points, a calibration matrix K_t and the motion matrix M_t, containing the pose parameters of the camera system, in

$$\mathbf{x}'_{it} = \lambda_{it}\,[\mathsf{K}_t \mid \mathbf{0}]\,\mathsf{M}_t^{-1}\mathbf{X}_i = \lambda_{it}\mathsf{P}_t\mathbf{X}_i\ . \tag{3.4}$$

This way, (a) homogeneous image coordinates allow for ideal image points, even directions opposite to the viewing direction, (b) homogeneous scene coordinates allow for far and ideal scene points, and including an additional motion is simply an additional factor.

However, this leads to two problems. As the covariance matrices $\Sigma_{\mathsf{x}'_{it}\mathsf{x}'_{it}}$ of homogeneous vectors are singular, the optimization function of the Maximum Likelihood Estimation

$$\sum_{it} \left\|\mathbf{x}'_{it} - \lambda_{it}\mathsf{P}_t\mathbf{X}_i\right\|^2_{\Sigma_{x'_{it}x'_{it}}} \tag{3.5}$$

is not valid. A minor, but practical problem is the increase of the number of unknown parameters, namely the Lagrangian multipliers, which are necessary when fixing the length of the vectors $\mathbf{X}_i$. In large bundle adjustments with more than a million scene points this prohibitively increases the number of unknowns by a factor 5/3.

3.1.5 Task and Challenges

The task is to model the projection process of a camera system as the basis for a bundle adjustment for a multi-view camera system, which (a) consists of mutually fixed single-view cameras, (b) allows the single cameras to be omnidirectional, requiring to explicitly model the camera rays and (c) which allows for far or ideal scene points for stabilizing the configuration. The model formally reads as

$$\mathscr{x}_{itc} = \mathcal{P}_c(\mathcal{M}_c^{-1}(\mathcal{M}_t^{-1}(\mathcal{X}_i))) \tag{3.6}$$

with the I scene points $\mathcal{X}_i, i = 1, ..., I$, the T motions $\mathcal{M}_t, t = 1, ..., T$ from the scene coordinates system into the individual camera coordinate systems, the C motions $\mathcal{M}_c$, $c = 1, ..., C$ of each single-view camera of the camera system, which make the mutual orientation explicit, and the projection $\mathcal{P}_c$ into the camera system $c = 1, ..., C$, and the observed image points $\mathscr{x}_{itc}$ of scene point i in camera c at time/pose t.

Figure 3.5 exemplary depicts a multi-camera system consisting of two fisheye cameras, which observes two scene points under two different poses. We will use the index t to denote the time of exposure, such that $\mathcal{M}_t$ describes the 6D pose of the multi-camera system at time t. Assuming the mutually relative poses between the cameras to be stable,

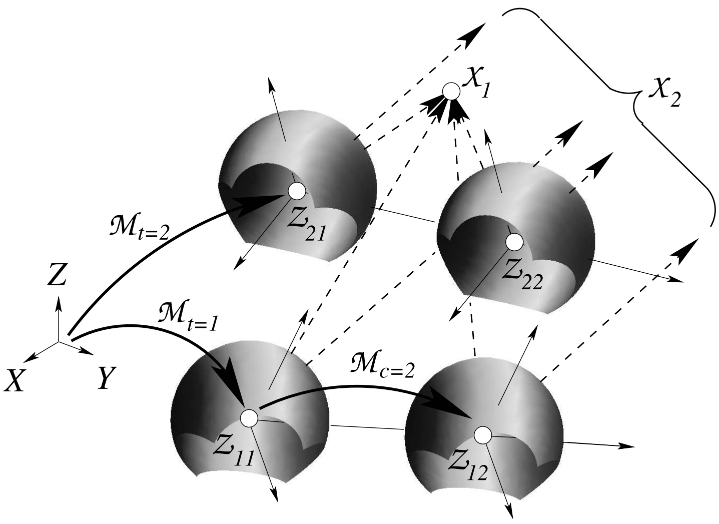


Figure 3.5: A two-camera system with fisheye cameras $c = 1, 2$ with projection centers $\mathcal{Z}_{tc}$ and known motion $\mathcal{M}_c$ and unknown motion $\mathcal{M}_t$, having a field of view larger than 180° shown at two exposure times $t = 1, 2$ observing two points $\mathcal{X}_i, i = 1, 2$, one being close, the other at infinity. Already a block adjustment with a single camera moving over time will be stabilized by points at infinity. Figure taken from Schneider et al. (2012).

the motion from the multi-camera system's coordinate system to each single camera with index c can be described with a 6D pose $\mathcal{M}_c$. In this example the reference system is identical to the camera system of the first camera $c = 1$. The two fisheye cameras observe two scene points on each camera position, $\mathcal{X}_1$ being close and $\mathcal{X}_2$ at infinity. Note that the intersection angles between the projection rays of point $\mathcal{X}_2$ are zero at each camera position as they intersect at infinity.

In order to realize this, we need to be able to represent bundles of rays together with their uncertainty, using uncertain direction vectors, to represent scene points at infinity using homogeneous coordinates, and minimize the number of parameters to be estimated. The main challenge lies in the inclusion of the statistics into an adequate minimal representation.

3.2 Related Work

Brown (1958) outlined the first approach to simultaneously adjust the entire set of observations of a photogrammetric net using a rigorous least squares adjustment to estimate point coordinates and camera positions. Subsequently, the various aspects of bundle adjustment have been studied intensively. Triggs et al. (2000) give a broad overview on bundle adjustment techniques in terms of estimation theory, robustification, solving large normal equation systems, gauge problems, outlier detection, network design and sensitivity analysis. Today, bundle adjustment is a central part of Structure from Motion (SfM) and visual SLAM systems, which showed remarkable success due to combining rigorous theory, advanced computational methods and a culture of open software development. Knapitsch et al. (2017) recently reviewed and benchmarked popular SfM pipelines and multi-view stereo techniques. SfM systems are designed to estimate camera motion and 3D structure of the environment in batch manner while visual SLAM is designed to run online. Taketomi et al. (2017) categorize recent visual SLAM algorithms into feature-based, direct and RGB-D camera-based approaches.

Multi-camera systems are proposed by many authors. Nistér et al. (2004) discuss the advantage of using a stereo video rig in order to avoid the difficulty with the scale transfer. Mostafa and Schwarz (2001) present an approach to integrate a multi-camera system with GPS and INS and discuss two approaches to calibrate such multi-sensor systems. Savopol et al. (2000) report on a multi-camera system for an aerial platform to increase the resolution and coverage of the terrain surface. Muhle et al. (2011) calibrate a multi-camera system with non-overlapping field of views by extracting the mutual orientation from a common motion. The authors examine critical motions and give a detailed accuracy analysis of the calibration, given different motion characteristics. Huang and Stachniss (2017) examine a rigorous motion-based calibration employing the Gauss–Helmert model and show the advantage over existing approaches in terms of accuracy. Carrera et al. (2011) calibrate a multi-camera rig with non-overlapping views using a SLAM approach to create for each camera a map of distinctive features. Finally, a global bundle adjustment is applied to estimate the relative poses of the cameras. Ly et al. (2014) use line features to calibrate the extrinsics of a multi-camera system by employing the unified projection model by Mei and Rives (2007). They show that an arbitrary camera can be used.

Pose estimation with a stereo rig is discussed in Hartley and Zisserman (2004, p. 493). Mouragnon et al. (2009) propose a bundle solution for stereo rigs working in terms of direction vectors, but they minimize the angular error without considering the covariance matrix of the observed rays. Frahm et al. (2004) present an approach for orienting a multi-camera system with non-overlapping views, however not applying a statistically rigorous approach. Bundle adjustment for multi-camera systems is extensively discussed in the thesis of Kim (2010). Nguyen and Lhuillier (2016) present a bundle adjustment for multi-camera systems consisting of consumer cameras. The proposed bundle adjustment is able to estimate the synchronization time offset between the single-view cameras and coefficients of the rolling shutter calibration but cannot handle all kinds of rotations because of singularity issues in the parametrization. Klingner et al. (2013) describe a SfM framework for generalized camera systems with rolling shutter cameras. Urban et al. (2017) present a bundle adjustment for multi-camera systems which includes the camera model of Scaramuzza et al. (2006), and allows for simultaneous self-calibration of the single-view cameras, but convergence is affected as shown by the authors. In all cases, points at infinity cannot be integrated.

Uncertain geometric reasoning using projective entities is extensively presented in Kanatani (1996), but only with normalized Euclidean geometric entities, which allows the estimation of some single geometric entities only. Heuel (2004), eliminating these deficiencies, proposes an estimation procedure which, however, does not eliminate the redundancy of the representation and also cannot easily include elementary constraints between observations, see Meidow et al. (2009). The developments made in in this section are based on the minimal representation schemes proposed in Förstner (2012), which reviews previous work and generalizes e.g. Bartoli (2002).

3.3 Model for a Moving Single-View Camera

We start by deriving a homogeneous expression to model the collinearity equations for image coordinates as observations, subsequently for ray directions as observations, and finally show how far and ideal scene points can be handled.

3.3.1 Image Coordinates as Observations

Using homogeneous coordinates

$$\mathbf{x}'_{it} = \lambda_{it} \mathsf{P}_t \mathbf{X}_i = \lambda_{it} \mathsf{K}_t R_t^{\mathsf{T}} \left[I_3 \mid -\boldsymbol{Z}_t \right] \mathbf{X}_i \tag{3.7}$$

with a projection matrix

$$\mathsf{P}_t = [\mathsf{K}_t \mid \mathbf{0}_3]\, \mathsf{M}_t^{-1}\,, \quad \mathsf{M}_t = \begin{bmatrix} R_t & \boldsymbol{Z}_t \\ \mathbf{0}_3^{\mathsf{T}} & 1 \end{bmatrix}$$

makes the motion of the camera explicit. It contains for each pose t the position of the projection center $\boldsymbol{Z}_t$ in the scene coordinate system and the rotation matrix R_t of the scene system to the camera system. The calibration matrix K_t contains parameters for the principal point, the principal distance, the affinity, and possibly lens distortion, see Förstner and Wrobel (2016, Eq. (12.61)) and Eq. (3.16). In case of an ideal camera with principal distance c, thus $\mathsf{K}_t = \mathrm{Diag}([c, c, 1])$, and Euclidean normalization of the homogeneous image coordinates with the k-th row $\mathbf{A}_{t,k}^{\mathsf{T}}$ of the projection matrix P_t

$$\mathbf{x}'^{e}_{it} = \frac{\mathsf{P}_t \mathbf{X}_i}{\mathbf{A}_{t,3}^{\mathsf{T}} \mathbf{X}_i} = \begin{bmatrix} \mathbf{A}_{t,1}^{\mathsf{T}} \mathbf{X}_i / \mathbf{A}_{t,3}^{\mathsf{T}} \mathbf{X}_i \\ \mathbf{A}_{t,2}^{\mathsf{T}} \mathbf{X}_i / \mathbf{A}_{t,3}^{\mathsf{T}} \mathbf{X}_i \\ 1 \end{bmatrix} \tag{3.8}$$

we obtain Eq. (3.2) and Eq. (3.3), i.e. $x'_{it} = \mathbf{A}_{t,1}^{\mathsf{T}} \mathbf{X}_i / \mathbf{A}_{t,3}^{\mathsf{T}} \mathbf{X}_i$ and $y'_{it} = \mathbf{A}_{t,2}^{\mathsf{T}} \mathbf{X}_i / \mathbf{A}_{t,3}^{\mathsf{T}} \mathbf{X}_i$.

Observe the transposition of the rotation matrix in Eq. (3.7), which differs from the notation used in several publications as Hartley and Zisserman (2004, Eq. (6.7)), but makes the motion of the camera from the scene coordinate system into the current camera system explicit, see Kraus et al. (2011).

3.3.2 Ray Directions as Observations

Using ray directions ${}^{c}\mathbf{x}'_{it}$, that point from camera t to scene point i and are oriented in the camera coordinate system, indicated by superscript k, we obtain the collinearity equations

$$\begin{aligned} {}^{c}\mathbf{x}'_{it} &= \lambda_{it}\,{}^{c}\mathsf{P}_t\mathbf{X}_i = \lambda_{it} R_t^{\mathsf{T}}(\boldsymbol{X}_i - \boldsymbol{Z}_t) \\ &= \lambda_{it}\,[I_3 \mid \mathbf{0}_3]\,\mathsf{M}_t^{-1}\mathbf{X}_i\,, \end{aligned} \tag{3.9}$$

where ${}^{c}\mathsf{P}_t$ does not contain the calibration matrix. Instead of Euclidean normalization, we now perform spherical normalization $\mathbf{x}^s = \mathsf{N}(\mathbf{x}) = \mathbf{x}/|\mathbf{x}|$ on both sides of the equation, in order to eliminate the scale factors λ_{it}. Spherical normalization yields the collinearity equations for camera bundles

$${}^{c}\mathbf{x}'^{s}_{it} = \mathsf{N}({}^{c}\mathsf{P}_t\mathbf{X}_i)\,. \tag{3.10}$$

We thus assume the camera bundles to be given as T sets $\{{}^{c}\mathbf{x}_{it}, i \in \mathcal{I}_t\}$ of normalized directions for each time t of exposure. The unknown parameters are the six parameters of the motion in ${}^{c}\mathsf{P}_t$ and the three independent parameters of each scene point $\mathbf{X}_i$.

Care has to be taken regarding the sign: We assume the negative ${}^{c}Z$-coordinate of the camera system to be the viewing direction. The scene points then need to have non-negative homogeneous coordinate $X_{i,4}$, which in case they are derived from Euclidean coordinates via $\mathbf{X}_i = [\boldsymbol{X}_i; 1]$ always is fulfilled. In case of ideal points, we therefore need to distinguish between the scene points $[\boldsymbol{X}_i; 0]$ and $[-\boldsymbol{X}_i; 0]$, which are points at infinity in opposite directions.

As a first result we observe that the difference between the classical collinearity equations and the collinearity equations for camera bundles is twofold. First, the unknown scale factor is eliminated differently: Euclidean normalization leads to the classical form in Eq. (3.8), spherical normalization leads to the bundle form in Eq. (3.10). Second, the calibration is handled differently: in the classical form it is made explicit, here we assume the image data to be transformed into camera rays taking the calibration into account. This will make a difference in modeling the individual cameras during self-calibration, a topic we will not discuss in this thesis.

3.3.3 Handling Far and Ideal Scene Points

Handling far and ideal scene points can easily be realized by also using *spherically* normalized coordinates $\mathbf{X}_i^s$ for the scene points leading to

$${}^{c}\mathbf{x}'^{s}_{it} = \mathsf{N}({}^{c}\mathsf{P}_t\mathbf{X}_i^s)\,. \tag{3.11}$$

Again care has to be taken regarding the sign of the points at infinity.

The confidence ellipsoid of 3D points can be used to visualize the achieved precision, in case the points are not too far. For a simultaneous visualization of confidence ellipsoids of 3D points which are close and far w.r.t. the origin one could perform a stereographic projection of the 3D-space into a unit sphere, i.e. $\boldsymbol{X} \mapsto \boldsymbol{X}/(1+|\boldsymbol{X}|)$ together with the transformation of the confidence ellipsoids. The relative poses of points close to the origin will then be preserved, far points will sit close to the boundary of the sphere. Their uncertainty in distance to the origin can then be inferred using their distance to the boundary of the sphere.

3.4 Model for a Moving Multi-camera System

With an additional motion

$$\mathsf{M}_c = \begin{bmatrix} R_c & \boldsymbol{Z}_c \\ \mathbf{0}_3^\mathsf{T} & 1 \end{bmatrix} \tag{3.12}$$

for each camera c of the camera system we obtain the general model for camera bundles

$${}^c\mathbf{x}'^s_{itc} = \mathsf{N}\left([I_3 \mid \mathbf{0}_3]\, \mathsf{M}_c^{-1}\, \mathsf{M}_t^{-1}\, \mathbf{X}_i^s\right) \ , \tag{3.13}$$

which makes all elements explicit: the observed directions $\mathcal{X}'_{itc}({}^c\mathbf{x}'_{itc})$ represented by normalized 3-vectors, having two degrees of freedom, unknown or known scene point coordinates $\mathcal{X}_i(\mathbf{X}_i^s)$, represented by spherically normalized homogeneous 4-vectors, having three degrees of freedom, unknown pose $\mathcal{M}_t(\mathsf{M}_t)$ of the camera system, having six parameters for each time a set of images was taken, and known or unknown calibration $\mathcal{M}_c(\mathsf{M}_c)$ containing the relative pose of the cameras, which are assumed to be rigid over time, having six parameters per camera. We will refer the relative poses to the first camera as reference camera with $R_{c=1} = I_3$ and $\boldsymbol{Z}_{c=1} = \mathbf{0}_3$.

Substituting

$${}^c\mathsf{P}_c = R_c^\mathsf{T}\, [I_3 \mid -\boldsymbol{Z}_c] = [I_3 \mid \mathbf{0}_3]\, \mathsf{M}_c^{-1} \tag{3.14}$$

yields the model

$${}^c\mathbf{x}'^s_{itc} = \mathsf{N}\left({}^c\mathsf{P}_c \mathsf{M}_t^{-1} \mathbf{X}_i^s\right) \ , \tag{3.15}$$

which we will use in case the system calibration is given, i.e. relative poses M_c are known.

3.5 Generating Camera Directions from Observed Image Coordinates

In most cases the observations are made using a digital camera whose sensor is approximately planar. The transition to directions of camera rays needs to be performed before starting the bundle adjustment. As mentioned before, this requires the internal camera geometry to be known. Moreover, in order to arrive at a statistically optimal solution, one needs to transfer the uncertainty of the observed image coordinates to the uncertainty of the camera rays. As an example we discuss two cases.

3.5.1 Perspective Cameras

In case we have perspective cameras with small image distortions, we can use the camera-specific and in general temporally varying calibration matrix

$$\mathsf{K}(\boldsymbol{x}', \boldsymbol{q}) = \begin{bmatrix} c & cs & x'_H + \Delta x(\boldsymbol{x}', \boldsymbol{q}) \\ 0 & c\,(1+m) & y'_H + \Delta y(\boldsymbol{x}', \boldsymbol{q}) \\ 0 & 0 & 1 \end{bmatrix} \tag{3.16}$$

for the forward transformation

$${}^{g}\mathbf{x}' = \mathsf{K}(\boldsymbol{x}', \boldsymbol{q})\, {}^{c}\mathbf{x}'^{s} \tag{3.17}$$

leading to the observable image coordinates ${}^{g}\mathbf{x}'$, the ^{g} indicates that the mapping can handle general distortions via additional parameters $\boldsymbol{q}$. Besides the basic parameters, namely the principal distance c with image plane ${}^{c}Z = c$, shear s, scale difference m, and principal point $\boldsymbol{x}'_H$, the calibration matrix contains additive corrections for modeling lens distortion or other deviations, which depend on the additional parameters $\boldsymbol{q}$ and spatially differ as a function of $\boldsymbol{x}$. In case of small deviations Eq. (3.17) can easily be inverted. However, one must take into account the different signs of the coordinate vector and the direction from the camera to the scene point, see Figure 3.6,

$${}^{c}\mathbf{x}'^{s} \approx \tau\, \mathsf{N}\left(\mathsf{K}^{-1}({}^{g}\boldsymbol{x}', \boldsymbol{q})\, {}^{g}\mathbf{x}'\right) \tag{3.18}$$

with $\tau \in \{-1, +1\}$ such that ${}^{c}x_3'^{s} < 0$. This relation is independent of the sign of the third element of the calibration matrix. Note that a point ${}^{g}\mathbf{x}'$ at infinity corresponds to the direction ${}^{c}\mathbf{x}'$ perpendicular to the viewing direction.

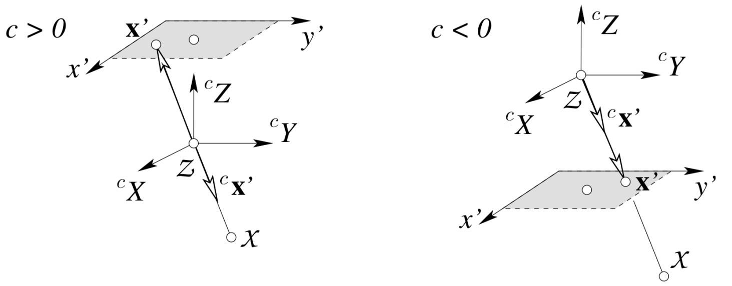


Figure 3.6: The direction of the homogeneous image coordinate vector and the direction of the ray is different depending on the sign of the principal distance c.

Given the covariance matrix $\Sigma_{{}^{g}\mathbf{x}'\,{}^{g}\mathbf{x}'}$ of the image point coordinates and the internal camera geometry with calibration matrix K, the covariance matrix of ${}^c\mathbf{x}'$ can be derived by variance propagation. Omitting the dependency of the calibration matrix on the point coordinates $\boldsymbol{x}'$, we have

$$\Sigma_{{}^c\mathrm{x}'{}^c\mathrm{x}'} = \mathsf{K}^{-1}\Sigma_{{}^g\mathrm{x}'{}^g\mathrm{x}'}\mathsf{K}^{-\mathsf{T}}\,. \tag{3.19}$$

We obtain the covariance matrix of the spherically normalized ray direction ${}^c\mathbf{x}'^s$ according to Eq. (2.4) by

$$\Sigma_{{}^c\mathrm{x}'^s{}^c\mathrm{x}'^s} = J_s({}^c\mathbf{x}')\Sigma_{{}^c\mathrm{x}'{}^c\mathrm{x}'}J_s^{\mathsf{T}}({}^c\mathbf{x}') \qquad \text{with} \qquad J_s(\mathbf{x}) = \frac{1}{|\mathbf{x}|}\left(I - \frac{\mathbf{x}\mathbf{x}^{\mathsf{T}}}{|\mathbf{x}|^2}\right). \tag{3.20}$$

3.5.2 Omnidirectional Single View Point Cameras

As an example for an omnidirectional single-view camera we take a camera with a fisheye lens. We model the projection of the fisheye lens with the equi-distant projection model introduced in Sec. 2.2.1. The interior orientation of a camera is determined separately by camera calibration according to Abraham and Hau (1997) using Chebyshev polynomials to describe distortion from the projection model. Using the equi-distant projection and applying all corrections, we obtain image points ${}^e\chi$ lying closer to the principal point $\mathcal{H}$ than the gnomonic projections ${}^g\chi$ of the scene points, see Figure 3.7. The ray direction ${}^c\mathbf{x}'^s$ can be derived from ${}^e\chi$ by using the normalized radial distance $r = |{}^e\boldsymbol{x}|$ growing with the angle ϕ between the viewing direction and the camera ray, see Eq. (2.20).

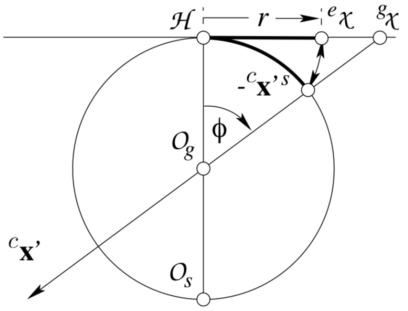


Figure 3.7: Relation between sensor point, viewing direction and viewing ray in the equi-distant projection model.

Again, the uncertainty of the image coordinates can be transformed to the uncertainty of the direction ${}^c\mathbf{x}'^s$ of the camera ray via variance propagation: Given the covariance matrix $\Sigma_{{}^e\mathrm{x}'{}^e\mathrm{x}'}$ of the image coordinates and principal distance c, we have

$$\Sigma_{{}^c\mathrm{x}'{}^c\mathrm{x}'} = \frac{1}{c^2}\, J_e^c({}^e\boldsymbol{x})\,\Sigma_{{}^e\mathrm{x}'{}^e\mathrm{x}'}\,J_e^c({}^e\boldsymbol{x})^{\mathsf{T}} \tag{3.21}$$

with

$$\mathsf{J}^c_e(\boldsymbol{x}) = \begin{bmatrix} s\mathsf{I}_2 + \dfrac{1}{|\boldsymbol{x}|}(\cos|\boldsymbol{x}| - s)\,\boldsymbol{x}\boldsymbol{x}^\mathsf{T} & \mathbf{0}_2 \\ s\boldsymbol{x}^\mathsf{T} & 0 \end{bmatrix}, \quad s = \frac{\sin|\boldsymbol{x}|}{|\boldsymbol{x}|}\,, \tag{3.22}$$

when using ${}^e\boldsymbol{x}$ in the inverse projection function in Eq. (2.20). Note that propagated covariance Eq. (3.21) depends on the image point position, unlike in Eq. (3.19). Accordingly, we obtain the covariance matrix of the spherically normalized ray direction ${}^c\mathbf{x}'^s$ via error propagation following Eq. (3.20).

In all cases the covariance matrix of the camera ray is singular as the normalized 3-vector only depends on two observed image coordinates.

3.6 The Estimation Procedure

We start with observed image points $\{\boldsymbol{x}, \Sigma_{xx}\}_{itc}$, which are assumed to be corrupted with mutually uncorrelated Gaussian noise, the coordinates of an image point, however, can be correlated. Image points are transfered into the corresponding camera directions

$$\{\mathbf{x}, \Sigma_{\mathrm{xx}}\}_{itc} := \{{}^c\mathbf{x}^s, \Sigma_{{}^c\mathrm{x}^s\,{}^c\mathrm{x}^s}\}_{itc} \tag{3.23}$$

as described in the previous section. To simplify readability, from now on all homogeneous vectors are assumed to be spherically normalized. Additionally, we will omit indices for the coordinate frame.

Formally, the task is to find best estimates $\widehat{\mathsf{M}}_t$ for all poses $t = 1, ..., T$ and $\widehat{\mathbf{X}}_i$ for all scene points $i = 1, ..., I$ that minimize the weighted reprojection errors

$$\Omega(\mathsf{M}_t, \mathbf{X}_i) = \sum_{itc} \left\| \mathbf{x}_{itc} - \mathsf{N}\left(\mathsf{P}_c\,\mathsf{M}_t^{-1}\,\mathbf{X}_i\right) \right\|_{\Sigma_{\mathrm{x}_{itc}\mathrm{x}_{itc}}} \tag{3.24}$$

with $\|\boldsymbol{a}\|_\Sigma = \boldsymbol{a}^\mathsf{T}\Sigma^{-1}\boldsymbol{a}$, in case the system calibration M_c in $\mathsf{P}_c = [\mathsf{I}_3 \mid \mathbf{0}_3]\,\mathsf{M}_c$ is known.

In case of a system self-calibration, the task is to additionally find best estimates $\widehat{\mathsf{M}}_c$ for relative poses $c = 2, ..., C$ besides best estimates $\widehat{\mathsf{M}}_t$ and $\widehat{\mathbf{X}}_i$ minimizing the weighted reprojection errors

$$\Omega(\mathsf{M}_c, \mathsf{M}_t, \mathbf{X}_i) = \sum_{itc} \left\| \mathbf{x}_{itc} - \mathsf{N}\left(\mathsf{P}_c\,\mathsf{M}_t^{-1}\,\mathbf{X}_i\right) \right\|_{\Sigma_{\mathrm{x}_{itc}\mathrm{x}_{itc}}}\,. \tag{3.25}$$

In both cases, the collinearity equations contain three equations per observed camera ray and four parameters for each scene point, though, both being unit vectors. Therefore the covariance matrices $\Sigma_{\mathrm{x}_{itc}\mathrm{x}_{itc}}$ are singular, thus the inversion is not possible.

Additionally, more than the necessary parameters are contained in the equations. Therefore we want to reduce the number of parameters to the necessary minimum. We do this after linearization.

3.6.1 Initial Values

The estimation starts with initial values for the poses

$$\widehat{\mathsf{M}}_t^a = \begin{bmatrix} \widehat{R}_t^a & \widehat{\boldsymbol{Z}}_t^a \\ \mathbf{0}^\mathsf{T} & 1 \end{bmatrix} \tag{3.26}$$

of the camera system given for every time of exposure $t = 1, ..., T$ with $\widehat{R}_t^a$ for the rotation matrix and $\widehat{\boldsymbol{Z}}_t^a$ for the position, and with initial values $\widehat{\mathbf{X}}_i^a$ for the $i = 1, ..., I$ spherically normalized scene points

$$\widehat{\mathbf{X}}_i^a = \mathsf{N}\left(\begin{bmatrix} \widehat{\boldsymbol{X}}_i^a \\ 1 \end{bmatrix}\right) = \begin{bmatrix} \widehat{\boldsymbol{X}}_{i0}^a \\ \widehat{X}_{ih}^a \end{bmatrix}, \tag{3.27}$$

and in case of a system self-calibration, additionally with approximate values for the relative poses $\widehat{\mathsf{M}}_c^a$, $c = 1, ..., C$, otherwise the relative poses need to be known.

Evaluating the functional model at the initial values yields initial observations $\widehat{\mathbf{x}}_{itc}^a$ and the additive 3D corrections $\widehat{\boldsymbol{v}}_{itc}^a$ for the observed ray directions

$$\widehat{\mathbf{x}}_{itc}^a = \mathbf{x}_{itc} + \widehat{\boldsymbol{v}}_{itc}^a = \mathsf{N}\left(\widehat{\mathsf{P}}_c^a \, {\widehat{\mathsf{M}}_t^a}^{-1} \widehat{\mathbf{X}}_i^a\right). \tag{3.28}$$

If the relative poses M_c are known, i.e. are not to be estimated, we formally have

$$\widehat{\mathbf{x}}_{itc}^a = \mathbf{x}_{itc} + \widehat{\boldsymbol{v}}_{itc}^a = \mathsf{N}\left(\mathsf{P}_c \, {\widehat{\mathsf{M}}_t^a}^{-1} \widehat{\mathbf{X}}_i^a\right). \tag{3.29}$$

3.6.2 Linearization and Update for Pose Parameters

Linearization of the non-linear model leads to a linear substitute model which yields correction parameters that allow to iteratively derive corrections for the initial values.

An initial motion matrix $\widehat{\mathsf{M}}^a$ will be corrected by multiplication with a small motion $\mathsf{M}(\widehat{\Delta\boldsymbol{m}})$, thus by

$$\widehat{\mathsf{M}} = \mathsf{M}(\widehat{\Delta\boldsymbol{m}})\, \widehat{\mathsf{M}}^a = \begin{bmatrix} R(\widehat{\Delta\boldsymbol{r}}) & \widehat{\Delta\boldsymbol{Z}} \\ \mathbf{0}_3^\mathsf{T} & 1 \end{bmatrix} \widehat{\mathsf{M}}^a , \tag{3.30}$$

where $\mathsf{M}(\widehat{\Delta\boldsymbol{m}})$ depends on a small 6D motion vector

$$\widehat{\Delta\boldsymbol{m}} = \begin{bmatrix} \widehat{\Delta\boldsymbol{r}} \\ \widehat{\Delta\boldsymbol{Z}} \end{bmatrix} \tag{3.31}$$

that is to be estimated. Note that we use the same concatenation as introduced in Sec. 2.1.3, Eq. (2.13).

The exponential form of a small motion matrix gives us the approximation

$$\mathsf{M}(\widehat{\Delta \boldsymbol{m}}) = \exp(\Delta\mathsf{M}(\widehat{\Delta \boldsymbol{m}})) \approx I_4 + \widehat{\Delta\mathsf{M}}(\widehat{\Delta \boldsymbol{m}}) \; . \tag{3.32}$$

with

$$\Delta\mathsf{M}(\widehat{\Delta \boldsymbol{m}}) = \begin{bmatrix} S(\widehat{\Delta \boldsymbol{r}}) & \Delta \boldsymbol{Z} \\ \mathbf{0}^\mathsf{T} & 0 \end{bmatrix} \tag{3.33}$$

such that we can rewrite Eq. (3.30) as

$$\widehat{\mathsf{M}} \approx \left(I_4 + \Delta\mathsf{M}(\widehat{\Delta \boldsymbol{m}}) \right) \widehat{\mathsf{M}}^a \; . \tag{3.34}$$

In order to estimate $\widehat{\Delta \boldsymbol{m}}$ we need to linearize the inverse motion matrix

$$\widehat{\mathsf{M}}^{-1} = \widehat{\mathsf{M}}^{a^{-1}} \mathsf{M}^{-1}(\widehat{\Delta \boldsymbol{m}}) \tag{3.35}$$

$$\approx \widehat{\mathsf{M}}^{a^{-1}} \left(I_4 - \Delta\mathsf{M}(\widehat{\Delta \boldsymbol{m}}) \right) . \tag{3.36}$$

To obtain the refined motion matrix $\widehat{\mathsf{M}}$, an estimated update $\widehat{\Delta \boldsymbol{m}}$ is applied to the initial motion matrix $\widehat{\mathsf{M}}^a$ with

$$\widehat{\mathsf{M}} = \begin{bmatrix} R(\widehat{\Delta \boldsymbol{r}}) & \widehat{\Delta \boldsymbol{Z}} \\ \mathbf{0}_3^\mathsf{T} & 1 \end{bmatrix} \widehat{\mathsf{M}}^a \; . \tag{3.37}$$

With an estimated rotation update $\widehat{\Delta \boldsymbol{r}}$, we obtain a valid rotation matrix $R(\widehat{\Delta \boldsymbol{r}})$ with the Cayley transformation in Eq. (2.8).

3.6.3 Reduced Coordinates and Update of Coordinates

The classical iteration scheme for fitted observations, omitting the indices, involves an additive correction $\widehat{\mathbf{x}} = \widehat{\mathbf{x}}^a + \widehat{\Delta \mathbf{x}}$. The additive correction involves three unknown corrections in $\widehat{\Delta \mathbf{x}}$, but each observed direction has only two degrees of freedom. Additionally, the additive correction does not preserve spherical normalization of the fitted observation of the ray direction.

As the differential corrections to directions live in the tangent space, we apply corrections to the unit vectors using reduced coordinates following Förstner (2012). We obtain reduced coordinates, say the 2D vector $\boldsymbol{x}_r$ of a spherically normalized 3D direction $\mathbf{x}$, in the two-dimensional tangent space

$$J_r(\widehat{\mathbf{x}}^a) := \left. \frac{\partial \mathbf{x}}{\partial \boldsymbol{x}_r} \right|_{\mathbf{x}=\widehat{\mathbf{x}}^a} = \underbrace{[\boldsymbol{s}, \boldsymbol{t}]}_{3\times 2} = \operatorname{null}(\widehat{\mathbf{x}}^{a\mathsf{T}}) \tag{3.38}$$

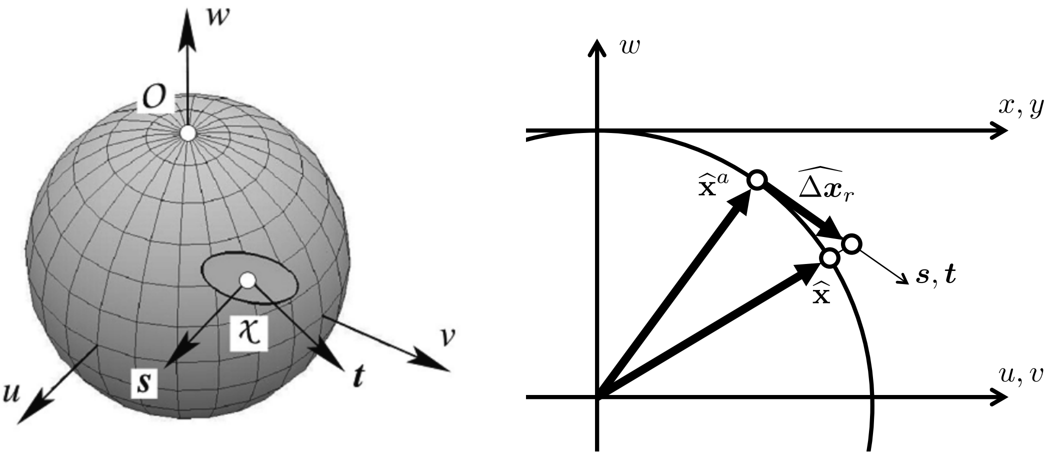


Figure 3.8: Left: Representation of a point with its uncertainty in the tangent space evaluated on the unit sphere. The uncertainty has only two degrees of freedom. Image adapted from Förstner (2012). Right: The estimated vector $\widehat{\mathbf{x}}$ is obtained by applying the estimated update $\widehat{\Delta \boldsymbol{x}_r}$ of approximate vector $\widehat{\boldsymbol{x}}^a$ in the tangent space $\mathrm{null}(\widehat{\mathbf{x}}^{a\mathsf{T}})$ followed by spherical normalization.

of the unit sphere S^2 evaluated at the approximate values $\widehat{\mathbf{x}}^a$, by

$$\boldsymbol{x}_r = \mathrm{null}^{\mathsf{T}}(\widehat{\mathbf{x}}^{a\mathsf{T}})\mathbf{x} = \begin{bmatrix} \boldsymbol{s}^{\mathsf{T}}\mathbf{x} \\ \boldsymbol{t}^{\mathsf{T}}\mathbf{x} \end{bmatrix}. \tag{3.39}$$

We now want to represent the uncertainty of $\boldsymbol{x}_r$ by a 2×2 matrix in that coordinate frame. This is easily achieved by projecting $\boldsymbol{\Sigma}_{\mathbf{xx}}$ of the spherically normalized ray direction into the tangent plane

$$\Sigma_{x_r x_r} = J_r^{\mathsf{T}}(\widehat{\mathbf{x}}^a)\Sigma_{\mathrm{xx}}J_r(\widehat{\mathbf{x}}^a) \tag{3.40}$$

resulting to a flat ellipsoid, see Figure 3.8(a). Between the uncertain reduced coordinates and its ray direction, we have the inverse relation

$$\Sigma_{\mathrm{xx}} = J_r(\widehat{\mathbf{x}}^a)\Sigma_{\mathrm{x}_r\mathrm{x}_r}J_r^{\mathsf{T}}(\widehat{\mathbf{x}}^a)\ . \tag{3.41}$$

We will estimate the updates $\widehat{\Delta \boldsymbol{x}_r}$ of these reduced coordinates. With estimated updates, and assuming spherically normalized homogeneous coordinates, this leads to the following update rule

$$\widehat{\mathbf{x}} = \mathsf{N}\left(\widehat{\mathbf{x}}^a + \mathrm{null}(\widehat{\mathbf{x}}^{a\mathsf{T}})\widehat{\Delta \boldsymbol{x}_r}\right) \tag{3.42}$$

visualized in Figure 3.8(b). Obviously, the initial vector $\widehat{\mathbf{x}}^a$ is updated by

$$\widehat{\Delta \mathbf{x}} = \mathrm{null}(\widehat{\mathbf{x}}^{a\mathsf{T}})\widehat{\Delta \boldsymbol{x}_r} \tag{3.43}$$

and then spherically normalized to achieve the updated direction vector $\widehat{\mathbf{x}}$ with unit length.

To linearize the coordinates of the scene points, we follow Eq. (3.43) and use the substitution

$$\widehat{\Delta\mathbf{X}}_i = \mathrm{null}(\widehat{\mathbf{X}}_i^{a\mathsf{T}})\,\widehat{\Delta\boldsymbol{X}}_{ri}\ . \tag{3.44}$$

With estimated 3-vector corrections $\widehat{\Delta\boldsymbol{X}}_{ri}$, which make the three degrees of freedom of each scene point explicit, we will apply the update rule in Eq. (3.42) to obtain corrected homogeneous scene point coordinates

$$\widehat{\mathbf{X}}_i^a = \mathsf{N}\left(\widehat{\mathbf{X}}_i^a + \mathrm{null}(\widehat{\mathbf{X}}_i^{a\mathsf{T}})\widehat{\Delta\boldsymbol{X}}_{ri}\right). \tag{3.45}$$

3.6.4 Linearized Model for Bundle Adjustment

In the following we will consider the case of a system self-calibration, i.e. when minimizing Eq. (3.25). With initial estimates for the parameters, Eq. (3.28) comprises a 3-vector residual $\widehat{\boldsymbol{v}}_{x_{itc}}$ for each ray direction. With reduced coordinates $\boldsymbol{x}_{ritc}$ of $\mathbf{x}_{itc}$ w.r.t. the initial camera direction $\widehat{\mathbf{x}}_{itc}^a$ we are able to reduce the number of equations per direction from three to two. Pre-multiplication of all observation equations with $J_r^{\mathsf{T}}(\widehat{\mathbf{x}}_{itc}^a)$ yields

$$J_r^{\mathsf{T}}(\widehat{\mathbf{x}}_{itc}^a)\,(\mathbf{x}_{itc} + \widehat{\boldsymbol{v}}_{\mathbf{x}_{itc}}^a) = J_r^{\mathsf{T}}(\widehat{\mathbf{x}}_{itc}^a)\,\mathsf{N}\left(\widehat{\mathsf{P}}_c^a\,\widehat{\mathsf{M}}_t^{a^{-1}}\widehat{\mathbf{X}}_i\right) \tag{3.46}$$

or with reduced coordinates

$$\boldsymbol{x}_{ritc} + \widehat{\boldsymbol{v}}_{x_{ritc}}^a = J_r^{\mathsf{T}}(\widehat{\mathbf{x}}^a)\,\mathsf{N}\left(\widehat{\mathsf{P}}_c^a\,\mathsf{M}_t^{a^{-1}}\widehat{\mathbf{X}}_i^a\right) \tag{3.47}$$

with 2-vector residuals $\widehat{\boldsymbol{v}}_{x_{ritc}}^a$ in the 2D tangent space evaluated at $\widehat{\mathbf{x}}_{itc}^a$, which make the two degrees of freedom of the observed directions explicit.

Each iteration solves for the 6D pose updates $\widehat{\Delta\boldsymbol{m}}_t$, the 6D pose updates $\widehat{\Delta\boldsymbol{m}}_c$, and 3D scene point coordinate updates $\widehat{\Delta\boldsymbol{X}}_{ri}$ of the linearized function

$$\begin{aligned}
\boldsymbol{x}_{ritc} + \widehat{\boldsymbol{v}}_{x_{ritc}}^a = \ & J_r^{\mathsf{T}}(\widehat{\mathbf{x}}_{itc}^a)\,\mathsf{N}\left(\widehat{\mathsf{P}}_c^a\,\widehat{\mathsf{M}}_t^{a^{-1}}\widehat{\mathbf{X}}_i^a\right)\\
& - J_r^{\mathsf{T}}(\widehat{\mathbf{x}}_{itc}^a)\,J_s(\widehat{\mathsf{P}}_c^a\,\widehat{\mathsf{M}}_t^{a^{-1}}\widehat{\mathbf{X}}_i^a)\,\widehat{\mathsf{P}}_c^a\,\Delta\mathsf{M}(\widehat{\Delta\boldsymbol{m}}_c)\,\widehat{\mathsf{M}}_t^{a^{-1}}\widehat{\mathbf{X}}_i^a\\
& - J_r^{\mathsf{T}}(\widehat{\mathbf{x}}_{itc}^a)\,J_s(\widehat{\mathsf{P}}_c^a\,\widehat{\mathsf{M}}_t^{a^{-1}}\widehat{\mathbf{X}}_i^a)\,\widehat{\mathsf{P}}_c^a\,\widehat{\mathsf{M}}_t^{a^{-1}}\Delta\mathsf{M}(\widehat{\Delta\boldsymbol{m}}_t)\,\widehat{\mathbf{X}}_i^a\\
& + J_r^{\mathsf{T}}(\widehat{\mathbf{x}}_{itc}^a)\,J_s(\widehat{\mathsf{P}}_c^a\,\widehat{\mathsf{M}}_t^{a^{-1}}\widehat{\mathbf{X}}_i^a)\,\widehat{\mathsf{P}}_c^a\,\widehat{\mathsf{M}}_t^{a^{-1}}\widehat{\Delta\mathbf{X}}_i
\end{aligned} \tag{3.48}$$

with

$$\Delta\mathsf{M}(\widehat{\Delta\boldsymbol{m}}) = \begin{bmatrix} S(\widehat{\Delta\boldsymbol{r}}) & \widehat{\Delta\boldsymbol{Z}} \\ \mathbf{0}^{\mathsf{T}} & 0 \end{bmatrix} \qquad \text{and} \qquad \widehat{\Delta\mathbf{X}}_i = J_r(\widehat{\mathbf{X}}_i^a)\widehat{\Delta\boldsymbol{X}}_{ri} \tag{3.49}$$

from Eq. (3.33) and Eq. (3.44).

In order to obtain the linearized functional model

$$\boldsymbol{x}_{ritc} + \widehat{\boldsymbol{v}}_{x_{ritc}} = C_{itc}^{\mathsf{T}} \widehat{\Delta \boldsymbol{X}}_{ri} + D_{itc}^{\mathsf{T}} \widehat{\Delta \boldsymbol{m}}_t + E_{itc}^{\mathsf{T}} \widehat{\Delta \boldsymbol{m}}_c \,, \tag{3.50}$$

which makes the 2×3 Jacobian matrix C_{itc}^{T} and 2×6 Jacobian matrices D_{itc}^{T} and E_{itc}^{T} explicit, we exploit the differential motion to obtain the expression

$$\Delta \mathsf{M}(\widehat{\Delta \boldsymbol{m}})\, \mathbf{X} = \begin{bmatrix} S(\widehat{\Delta \boldsymbol{r}}) & \widehat{\Delta \boldsymbol{Z}} \\ \mathbf{0}^{\mathsf{T}} & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{X}_0 \\ X_h \end{bmatrix} \tag{3.51}$$

$$= \begin{bmatrix} -S(\boldsymbol{X}_0) & X_h I_3 \\ \mathbf{0}^{\mathsf{T}} & 0 \end{bmatrix} \begin{bmatrix} \widehat{\Delta \boldsymbol{r}} \\ \widehat{\Delta \boldsymbol{Z}} \end{bmatrix} \tag{3.52}$$

with the Euclidean part $\boldsymbol{X}_0$ and homogeneous part X_h of scene point $\mathbf{X} = \left[\boldsymbol{X}_0^{\mathsf{T}}, X_h\right]^{\mathsf{T}}$, and simplify with inverse length $p_{itc} = 1/|\widehat{\mathsf{P}}_c^a \, \widehat{\mathsf{M}}_t^{a^{-1}} \widehat{\mathbf{X}}_i^a|$ of the projected non-normalized direction

$$J_r^{\mathsf{T}}(\widehat{\mathbf{x}}_{itc}^a)\, J_s(\widehat{\mathsf{P}}_c^a \, \widehat{\mathsf{M}}_t^{a^{-1}} \widehat{\mathbf{X}}_i^a) = p_{itc} J_r^{\mathsf{T}}(\widehat{\mathbf{x}}_{itc}^a) \tag{3.53}$$

as $J_s(\mathbf{x})$ can be rewritten as

$$J_s(\mathbf{x}) = \frac{1}{|\mathbf{x}|} J_r(\mathbf{x}) J_r^{\mathsf{T}}(\mathbf{x}) \qquad \text{and} \qquad J_r^{\mathsf{T}}(\mathbf{x}) J_r(\mathbf{x}) = I_2 \,. \tag{3.54}$$

These reformulations lead to the explicit Jacobians

$$C_{itc}^{\mathsf{T}} = p_{itc} J_r^{\mathsf{T}}(\widehat{\mathbf{x}}_{itc}^a)\, \widehat{\mathsf{P}}_c^a \, \widehat{\mathsf{M}}_t^{a^{-1}} J_r(\widehat{\mathbf{X}}_i^a) \tag{3.55}$$

$$D_{itc}^{\mathsf{T}} = -p_{itc} J_r^{\mathsf{T}}(\widehat{\mathbf{x}}_{itc}^a)\, \widehat{\mathsf{P}}_c \, \widehat{\mathsf{M}}_t^{a^{-1}} \begin{bmatrix} -S(\widehat{\boldsymbol{X}}_0^a) & \widehat{X}_h^a I_3 \\ \mathbf{0}^{\mathsf{T}} & 0 \end{bmatrix} \tag{3.56}$$

$$= -p_{itc} J_r^{\mathsf{T}}(\widehat{\mathbf{x}}_{itc}^a)\, \widehat{R}_c^{a^{\mathsf{T}}} \, R_t^{\mathsf{T}} \left[-S(\widehat{\boldsymbol{X}}_0^a) \mid \widehat{X}_h^a I_3 \right] \tag{3.57}$$

$$E_{itc}^{\mathsf{T}} = -p_{itc} J_r^{\mathsf{T}}(\widehat{\mathbf{x}}_{itc}^a)\, \widehat{\mathsf{P}}_c^a \begin{bmatrix} -S\left(\widehat{R}_t^{a^{\mathsf{T}}} (\widehat{\boldsymbol{X}}_0^a - \widehat{X}_0^a \widehat{\boldsymbol{Z}}_t^a) \right) & \widehat{X}_h^a I_3 \\ \mathbf{0}^{\mathsf{T}} & 0 \end{bmatrix} \tag{3.58}$$

$$= -p_{itc} J_r^{\mathsf{T}}(\widehat{\mathbf{x}}_{itc}^a)\, \widehat{R}_c^{a^{\mathsf{T}}} \left[-S(\widehat{R}_t^{a^{\mathsf{T}}} (\widehat{\boldsymbol{X}}_0^a - \widehat{X}_0^a \widehat{\boldsymbol{Z}}_t^a)) \mid \widehat{X}_h^a I_3 \right] . \tag{3.59}$$

We now arrive at a well-defined optimization problem: find all $\widehat{\Delta \boldsymbol{X}}_{ri}$, $\widehat{\Delta \boldsymbol{R}}_t$, $\widehat{\Delta \boldsymbol{Z}}_t$, $\widehat{\Delta \boldsymbol{R}}_c$, $\widehat{\Delta \boldsymbol{Z}}_c$ minimizing

$$\Omega\left(\widehat{\Delta \boldsymbol{X}}_{ri}, \widehat{\Delta \boldsymbol{R}}_t, \widehat{\Delta \boldsymbol{Z}}_t, \widehat{\Delta \boldsymbol{R}}_c, \widehat{\Delta \boldsymbol{Z}}_c \right) = \sum_{itc} \widehat{\boldsymbol{v}}_{x_{ritc}}^{\mathsf{T}} \Sigma_{x_{ritc} x_{ritc}}^{-1} \widehat{\boldsymbol{v}}_{x_{ritc}} \tag{3.60}$$

with the regular 2×2 covariance matrices $\Sigma_{x_{ritc} x_{ritc}}$. The optimization problem can be solved iteratively following the estimation procedure of the non-linear Gauss–Markov Model in Sec. 2.3.1 with the update rules for scene point coordinates in Eq. (3.45) and motion matrices in Eq. (3.30).

In case the relative poses M_c are assumed to be known, we formally have the optimization problem

$$\Omega\left(\widehat{\Delta \boldsymbol{X}}_{ri}, \widehat{\Delta \boldsymbol{R}}_t, \widehat{\Delta \boldsymbol{Z}}_t\right) = \sum_{itc} \widehat{\boldsymbol{v}}^{\mathsf{T}}_{x_{ritc}} \Sigma^{-1}_{x_{ritc}x_{ritc}} \widehat{\boldsymbol{v}}_{x_{ritc}} \,. \tag{3.61}$$

The final Euclidean 3D coordinates $\widehat{\boldsymbol{X}}_i$ of scene points that are not at infinity with covariance matrices $\Sigma_{\widehat{X}_i\widehat{X}_i}$ are obtained by

$$\widehat{\boldsymbol{X}}_i = \frac{\widehat{\boldsymbol{X}}_{i,0}}{\widehat{X}_{i,h}} \qquad \text{and} \qquad \Sigma_{\widehat{X}_i\widehat{X}_i} = J_{\widehat{X}_i\widehat{\Delta X}_{ri}} \Sigma_{\widehat{\Delta X}_{ri}\widehat{\Delta X}_{ri}} J^{\mathsf{T}}_{\widehat{X}_i\widehat{\Delta X}_{ri}} \tag{3.62}$$

with Jacobian matrix

$$J_{\widehat{X}_i\widehat{\Delta X}_{ri}} = \frac{1}{\widehat{X}^2_{i,h}} \left[\widehat{X}_{i,h} I_3 \mid \widehat{\boldsymbol{X}}_{i,0}\right] \text{null}(\widehat{\mathbf{X}}_i^{\mathsf{T}}) \,. \tag{3.63}$$

3.7 Experiments

The approach for the rigorous bundle adjustment for omnidirectional and multi-view cameras described above has been implemented and tested on datasets gathered with real multi-camera systems and a simulated camera system. The experiments are designed to check the correctness of the implemented model and the advantage of including far points or points with glancing intersections within the bundle adjustment.

We first give some details on the implementation of the bundle adjustment used in the experiments. Subsequently, we check the correctness and feasibility of the implemented model and investigate the decrease of the precision when excluding far points in bundle adjustment. Finally we will evaluate the approach to calibrate multi-camera systems in a system self-calibration.

The interior orientation of each camera has been recovered by camera calibration as detailed in Sec. 2.2.2.

3.7.1 Implementation Details

So far, we have described a free bundle adjustment without control information or additional constraints to define the gauge. As a consequence, the coordinate system can be chosen freely up to a similarity transformation leading to a rank deficiency of seven when solving for the unknown parameters. For the similarity model we need to fix the position of the origin, the direction of the axes, and the overall scale. We enforce seven centroid constraints on the scene point coordinates in Euclidean space, thus we transform the reduced corrections $\widehat{\Delta \boldsymbol{X}}_{ri}$ into projective space and accordingly we compute the difference between initial and updated scene points in Euclidean space. Those constraints can only be applied on scene points that are not at infinity. The explicit Jacobian of the seven linearized centroid constraints on the reduced coordinate updates $\widehat{\Delta \boldsymbol{X}}_{ri}$ reads as

$$\sum_{\{i \in I \neg I_\infty\}} \begin{bmatrix} \mathsf{I}_3 \\ \mathsf{S}^{\mathsf{T}}(\widehat{\boldsymbol{X}}_i^a) \\ \widehat{\boldsymbol{X}}_i^{a\mathsf{T}} \end{bmatrix} \mathsf{J}_d(\widehat{\mathbf{X}}_i^a) \mathsf{J}_r(\widehat{\mathbf{X}}_i^a) \widehat{\Delta \boldsymbol{X}}_{ri} = 0 \tag{3.64}$$

with

$$\mathsf{J}_r(\mathbf{X}) = \operatorname{null}(\mathbf{X}^{\mathsf{T}}) \qquad \text{and} \qquad \mathsf{J}_d(\mathbf{X}) = \frac{1}{X_h^2} \left[X_h \mathsf{I}_3 \mid -\boldsymbol{X}_0 \right] . \tag{3.65}$$

Imposing the centroid constraints results in a free bundle adjustment, in which the trace of the covariance matrix of the estimated scene points is minimal.

Using multi-camera systems the scale is in fact defined by the known translations $\boldsymbol{Z}_c$ between the single-view cameras. However, the spatial extent of the whole block can be very large compared to the magnitude of this translation leading to an ill-posed normal equation system. If the spatial extent of the block is large, we consider this by applying a crisp constraint on the scale as formulated in Eq. (3.64). If the spatial extent is small, we make the constraint on the scale weak by declaring it to a stochastic observation with some covariance which allows the constraint to deviate from zero.

For initialization sufficiently accurate initial values for scene point coordinates $\widehat{\mathbf{X}}_i^a$ and for translation and rotation of the camera system poses $\widehat{\mathsf{M}}_t^a$ at times of synchronized exposure are needed. For the real dataset acquired with a multi-camera system, first, we determine the pose of each camera without considering the cameras as a rigid multi-camera rig with the SIFT-feature based bundle adjustment aurelo provided by Läbe and Förstner (2006) and use the results as initial values for the pose of the multi-camera system. Scene points are triangulated with the use of all corresponding image points that are consistent with the estimated relative orientations.

We robustify the cost function by down weighting measurements whose residual errors are too large by minimizing the robust Huber cost function according to Huber (1981). The iterative Levenberg-Marquardt algorithm can be exploited as in (Lourakis and Argyros, 2009) to obtain a damped convergence, but the Gauss–Newton algorithm without regularization is used in the following.

3.7.2 Test on Correctness and Feasibility

We now employ a simulated scenario to check the correctness of the implemented model, and a dataset gathered with a consumer-grade stereo camera to check the feasibility on real data.

Simulated scenario. We simulated a multi-camera system moving on a rounded square, observing 50 close scene points and 10 scene points far away at the horizon, i.e. at infinity,

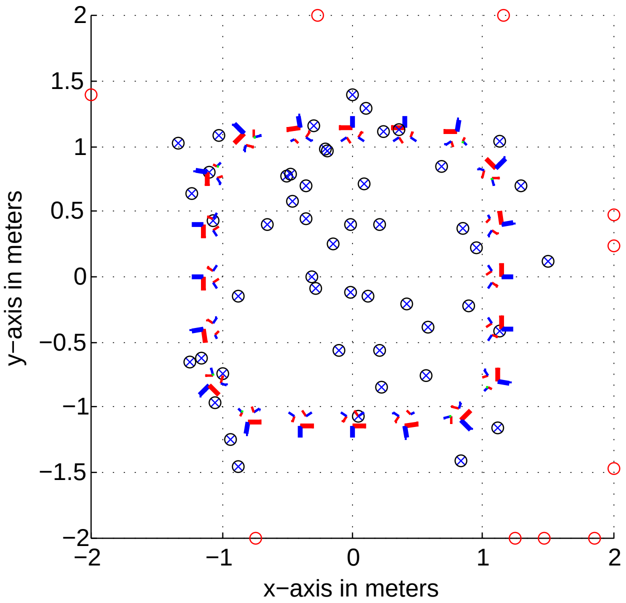


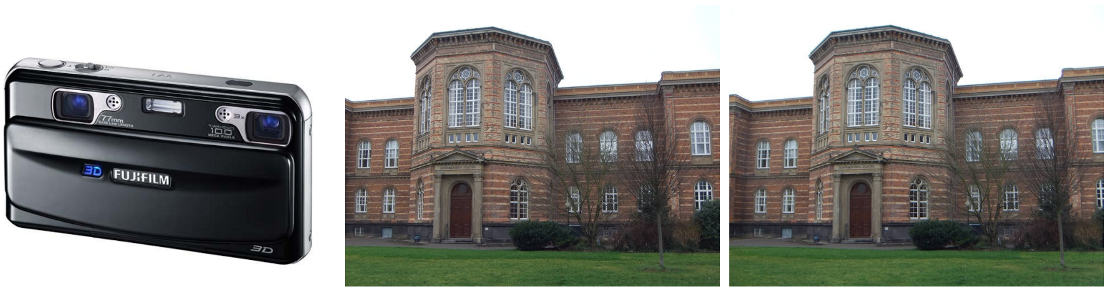
Figure 3.9: Simulation of a moving multi-camera system (poses of reference camera shown as bold tripods) with loop closing. Scene points nearby (crossed dots) and at the horizon (empty dots) being numerically at infinity are observed.

see Figure 3.9. The multi-camera system contains three single-view cameras. Every scene point is observed by a camera ray from all 20 positions of the camera system. The simulated set-up provides a high redundancy of observations.

Assuming the standard deviation of an image coordinate to be 0.3 pixel and a principal distance of 500 pixel, we add normally distributed noise with $\sigma_l = 0.3/500\,\mathrm{rad}$ on the spherically normalized camera rays to simulate the observation process. In order to obtain initial values for the bundle adjustment, we randomly disturb both, the generated spherical normalized homogeneous scene points $\mathbf{X}_i^s$, which are 4D-directions, by 6 °, and the generated motion parameters R_t and $\boldsymbol{Z}_t$ of each camera pose M_t by 3° and 10 % of the relative distances between the projection centers.

The iterative estimation procedure stops after six iterations, when the maximum normalized observation update is less than 10^{-6}. The residuals of the observed image rays in the tangent space of the adjusted camera rays, which are approximately angles between the rays in radiants, do not show any deviation from the normal distribution. The estimated a posteriori variance factor $\widehat{\sigma}_0^2 = 1.0021^2$ confirms the a priori stochastic model with variance factor $\sigma_0^2 = 1$.

In order to test if the estimated orientation parameters and scene point coordinates represent the maximum likelihood estimates for normally distributed noise on the observations, we have generated the same simulation 2,000 times with different random noise. The estimated variance factor is Fisher distributed but for high redundancy the distribution takes the shape of the normal distribution. The mean of the estimated variance factors is not significantly different from one and the theoretical covariance matrix does



(a) Finepix Real 3D W1[12] (b) left image (c) right image

Figure 3.10: Sample images of the stereo camera dataset.

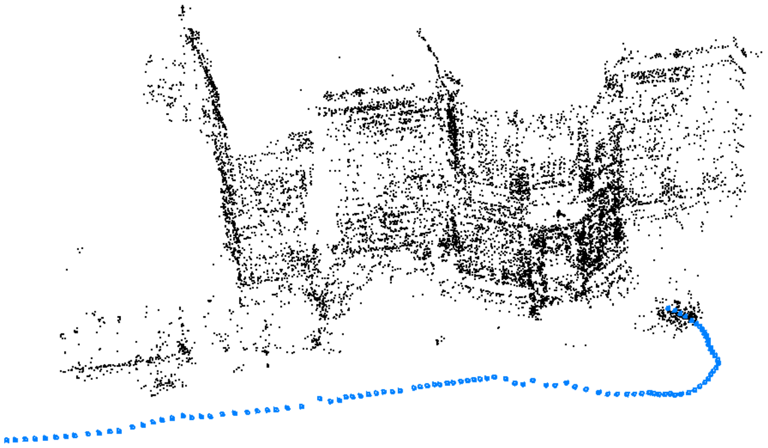


Figure 3.11: Illustration of the estimated scene points and poses of the dataset gathered with the FinePix Real 3D W1.

not differ significantly from the empirical covariance matrix according to the test statistic of Förstner and Wrobel (2016, Eq. (4.358), p. 140). These results confirm the correctness of the approach and that we can rely on the theoretical covariance matrix provided by the implemented estimation procedure.

Stereo Camera Dataset. In order to test the feasibility on real data, we apply the bundle adjustment on 100 stereo images of a building with a highly textured facade, taken with the consumer stereo camera FinePix Real 3D W1 from Fujifilm, see Figure 3.10. We use aurelo without considering the known relative orientation between the stereo images to obtain an initial solution for the camera poses and the scene points. The dataset contains 284,813 image points and 12,439 observed scene points.

Starting from an a priori standard deviation for the image coordinates of $\sigma_l = 1$ pixel the a posteriori variance factor is estimated with $\widehat{\sigma}_0 = 0.37$ indicating the automatically extracted SIFT points to have an average precision of approximately 0.4 pixel. The estimated scene points and poses are shown in Figure 3.11.

---

[12] Fujifilm Holdings K.K., Finepix Real 3D W1, digital image, accessed on 14 July 2018, `https://www.fujifilm.eu/fileadmin/product_migration/dc/headerimage/Finepix_Real_3D_W1_51.png`

3.7.3 Decrease of Rotational Precision Excluding Far Points

Classical bundle adjustment approaches, as used by aurelo, cannot handle scene points with glancing intersections – e.g. with maximal intersection angles lower than $\gamma = 1\,\text{gon}$ – which therefore are excluded in the estimation process to avoid numerical difficulties. Far scene points, however, can be observed over long periods of time and therefore should improve the quality of rotation estimation significantly. We investigate the decrease of precision L of the estimated rotation parameters of $\widehat{R}_t$ when excluding scene points with glancing intersection angles. In detail, we will determine the average empirical standard deviation

$$\sigma_{\alpha_t} = \widehat{\sigma}_0 \sqrt{\frac{1}{3} \operatorname{tr} \Sigma_{\widehat{r}_t \widehat{r}_t}} \tag{3.66}$$

for all estimated rotation parameters and report the average decrease of precision L by excluding far points determined by the geometric mean, namely

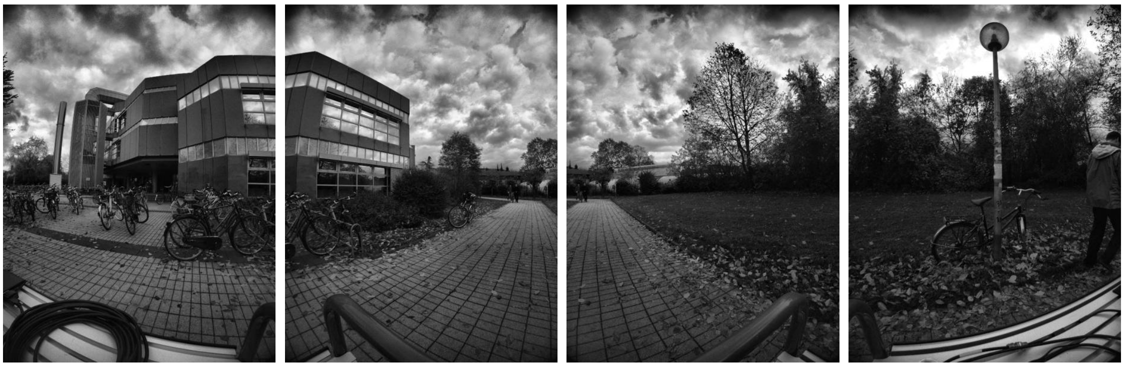
$$L = \exp\left[\sum_t^T \log(\sigma'_{\alpha_t}/\sigma_{\alpha_t})/T\right] \ , \tag{3.67}$$

where σ'_{α_t} represents the resulting average empirical standard deviation when scene points whose maximal intersection angle is lower than a threshold γ are excluded.

Simulated Scenario. We determine the decrease of precision L for the estimated rotation parameters by excluding a varying number of scene points at infinity on the basis of the simulation of a moving multi-camera system introduced in the previous section. Again we generate 50 scene points close to the multi-camera positions and vary the number of scene points at infinity to be 5, 10, 20, 50 and 100. The resulting average decrease in precision of the estimated rotations in $\widehat{\mathsf{M}}_t$ is 7.15 %, 11.77 %, 27.67 %, 54.56 % and 91.28 %, respectively. This strongly proves that points at infinity have a highly relevant positive influence on the rotational precision.

Multi-Camera Dataset. We apply the bundle adjustment to an image sequence consisting of 360 images taken by four of the six cameras of the multi-camera system Ladybug 3 shown in Figure 3.12. The Ladybug 3 is mounted on a hand-guided platform and is triggered once per meter using an odometer. The 90 m long trajectory of the Ladybug 3 consists of motion around a building. Initial values are obtained with aurelo by combining the individual cameras into a single virtual camera by adding corrections to the observed camera rays, which depend on the distance to the observed scene point as in Schmeing et al. (2011).

The dataset contains 10,891 of 26,890 scene points that are observed with a maximum intersection angle lower than $\gamma = 1\,\text{gon}$. The histogram in Figure 3.13 (a) shows the distribution of the maximum intersection angles per scene point. We examine the average



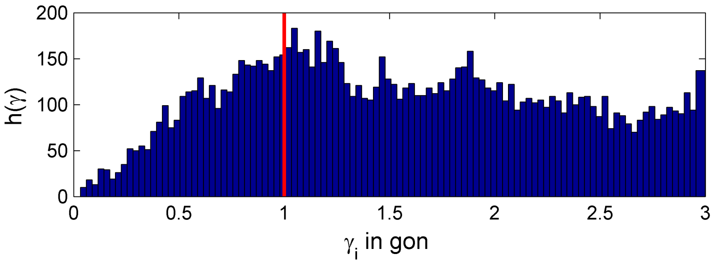
(a) cam 1 (b) cam 2 (c) cam 3 (d) cam 4

Figure 3.12: Sample images of the Ladybug 3 dataset. The dataset contains 360 images acquired on 90 camera positions on which four images have been taken with the same time of exposure.

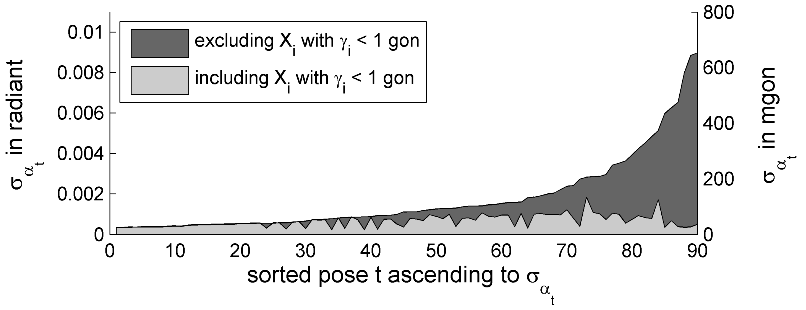
standard deviation σ_{α_t} of the estimated rotation parameters for two cases: in the first case, we exclude scene points with maximum intersection angle $\gamma < 1$ gon and in the second case, we include all scene points in the bundle adjustment. The average standard deviation σ_{α_t} of the estimated rotation parameters of each camera pose is shown in Figure 3.13 (b) for both cases. Some of the cameras show very large differences in precision, demonstrating the relevance of the far scene points in the Ladybug 3 dataset. The use of far points results in an almost constant precision of the rotation parameters over all camera poses, in contrast to the results of the bundle adjustment excluding far points. The individual gain in precision is mainly obtained due to a higher number of observed scene points at the individual poses, as can be seen in the scatter plot in Figure 3.13 (c). The estimated a posteriori variance factor amounts to $\widehat{\sigma}_0^2 = 1.05^2$ using an a priori stochastic model with $\sigma_l = 1$ pixel for the image points, indicating a quite poor precision of the point detection, which mainly results from the limited image quality.

Urban Drive Dataset. We make the same investigation on an image sequence consisting of 283 images taken by a single-view camera mounted on a car. The camera's viewing direction is aligned nearly orthogonal to the driving direction for the acquisition of building facades. The trajectory of the single-view camera consists of several turns at urban intersections in a residential area. The image sequence consists of 283 images. Corresponding image points and initial values for the camera poses and scene points are obtained with aurelo.

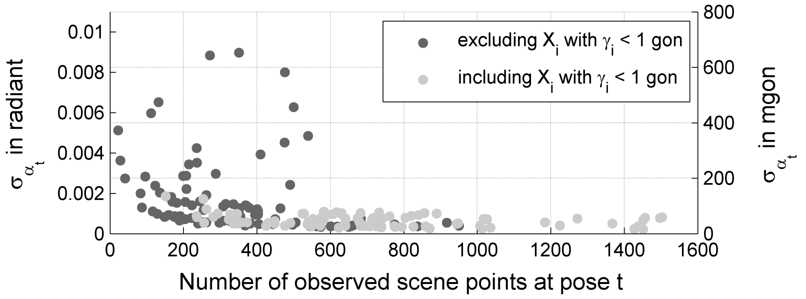
The dataset contains 33,274 of 62,401 scene points observed with a maximum intersection angle per point smaller than $\gamma = 1$ gon, see Figure 3.14 (a). Excluding those scene points decreases the average precision of the estimated rotation parameters by about 17.41 %. The average standard deviation σ_{α_t} of the estimated rotation parameters of each camera pose is shown in Figure 3.14 (b) showing the individual gain in precision that again is mainly obtained due to a higher number of observed scene points at the individual poses,



(a) Number of scene points with small intersection angles.



(b) Average empirical standard deviation of estimated rotations.

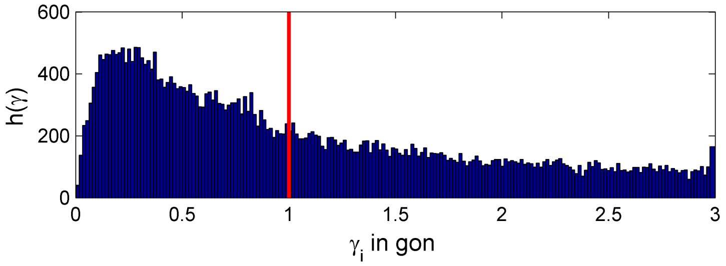


(c) Scatter plot of σ_{α_t} against the number of observed scene points at t

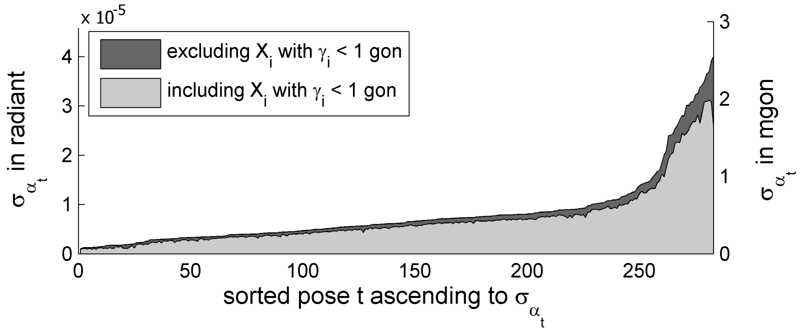
Figure 3.13: The histogram in (a) shows the number of scene points in the multi-camera dataset with small intersection angles. The average precision σ_{α_t} determined by excluding and including scene points with $\gamma < 1\,\text{gon}$ for all poses $t = 1, ..., T$ is compared to each other in (b) and against the number of observed scene points in (c).

shown in the scatter plot of Figure 3.14 (c). The estimated a posteriori variance factor amounts to $\widehat{\sigma}_0^2 = 0.54^2$ using an a priori stochastic model with $\sigma_l = 1\,\text{pixel}$ for the image points, indicating the precision to be in a normal range.

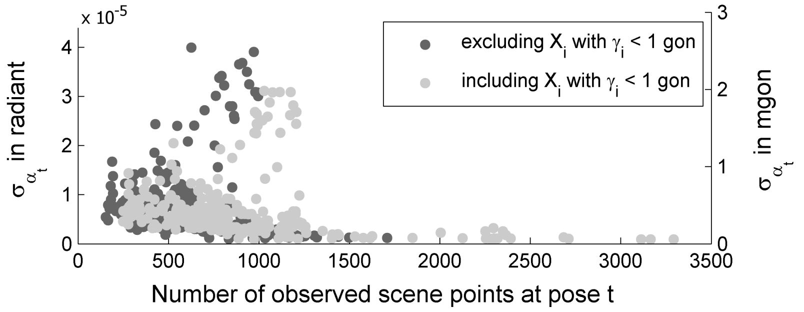
In contrast to the multi-camera dataset, the inclusion of scene points with small intersection angles does not result in an almost constant precision of the rotation parameters over all camera poses. Unlike the multi-camera dataset, the urban drive dataset does not benefit from far points which have been observed over multiple images. The tracks in the urban drive dataset are quite short, due to the alignment of the camera towards the building facades. However, the many scene points with small intersection angles can increase the precision of the rotation parameters with an almost constant factor on each camera pose.



(a) Number of scene points with small intersection angles.



(b) Average empirical standard deviation of estimated rotations.



(c) Scatter plot of σ_{α_t} against the number of observed scene points at t

Figure 3.14: The diagrams (a), (b) and (c) show the results of the evaluation of the urban drive dataset in the same way as Figure 3.13 for the multi-camera dataset.

3.7.4 Calibration of Multi-Camera Systems

So far, we have assumed the mutually relative poses between the cameras of the multi-camera system to be known. In case the relative poses M_c are unknown, the approach allows to perform a system self-calibration to additionally find best estimates $\widehat{\mathsf{M}}_c$ for relative poses besides best estimates $\widehat{\mathsf{M}}_t$ and $\widehat{\mathbf{X}}_i$.

In the following, we illustrate the calibration of three multi-camera systems and report the achieved precision of the estimated calibration. The camera systems differ in the configuration of the single-view cameras and the type of lenses.

Calibration with overlapping views. We now describe the calibration of the camera system shown in Figure 3.15 with highly overlapping views, which is used for 3D reconstruction of vines. In order to determine the relative poses of the multi-camera system,

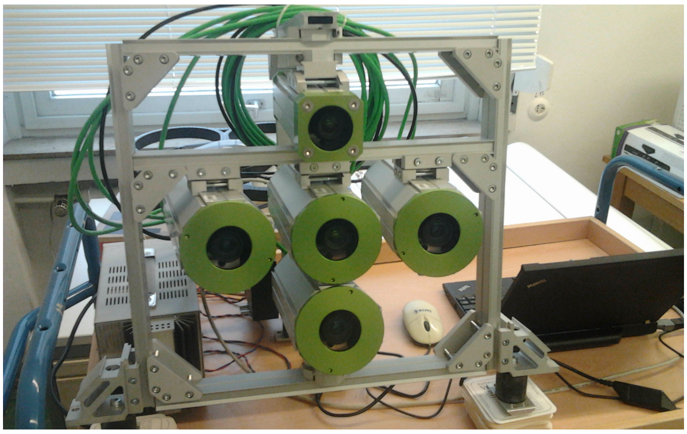
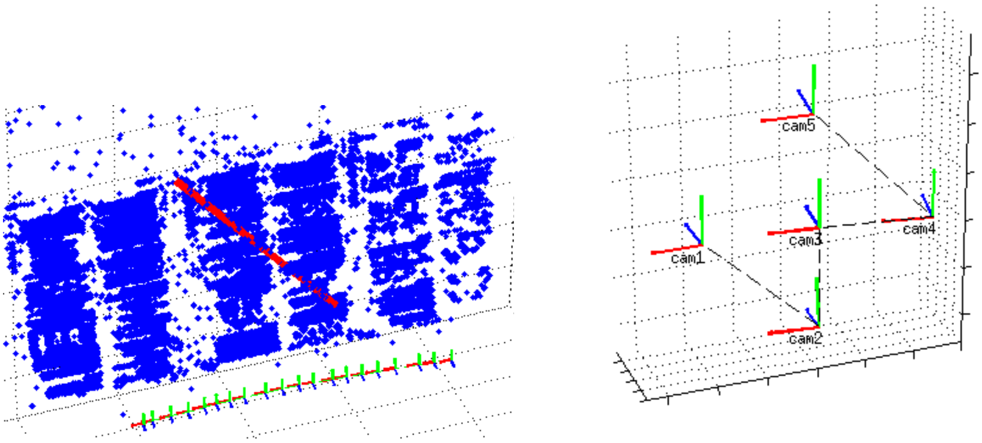


Figure 3.15: Multi-camera system consisting of five overlapping perspective camera views: Infrared camera on top, RGB camera in the middle and three monochromatic cameras. The distances from the RGB camera to the others are about 10 cm.



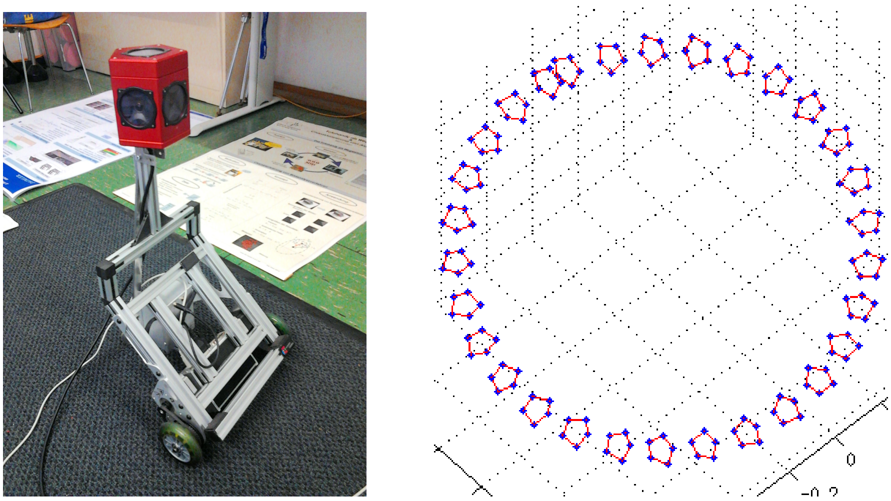
(a) (b)

Figure 3.16: Illustration of the estimated scene points and poses of the reference camera in (a). The red line denotes the known length on a poster for scale definition. The estimated relative poses of the multi-camera system are shown in (b).

we apply the bundle adjustment to 100 images of a wall draped with highly textured posters. The images were taken with the camera system from 20 different perspectives in a synchronized way. We use aurelo without considering the mutual stable relative poses between the cameras to obtain an initial solution for all 100 camera poses and the scene points. The dataset contains 593,412 observed image points of 63,140 scene points.

Starting from an a priori standard deviation for the image coordinates of $\sigma_l = 1$ pel, the a posteriori variance factor is estimated with $\widehat{\sigma}_0^2 = 0.11^2$ indicating the automatically extracted SIFT points to have an average precision of approximately 0.1 pel. This high precision of the point detection results mainly from the good image and camera calibration quality, and the highly distinctive scene structure. Figure 3.16 illustrates the estimated scene points and poses as well as the estimated relative poses.

The estimated uncertainty of the estimated rotations of the cameras regarding the ref-



(a) Ladybug 3 on robot. (b) Estimated camera poses.

Figure 3.17: The Ladybug 3 mounted on a robot (a) executing a circular movement. The poses given by aurelo of each camera are shown in (b). Note that aurelo applies no constraints for a rigid camera system and the scale is chosen arbitrarily.

erence camera is 0.1–0.2 mdeg around the viewing direction and 0.4–0.8 mdeg orthogonal to it. We scale the photogrammetric model by using a measured distance of 1.105 m with an error of about 0.1 %. The uncertainty of the estimated relative translations is 0.02–0.04 mm in viewing direction and 0.1–0.2 mm orthogonal to it.

Multi-camera system Ladybug 3. The omnidirectional multi-camera system Ladybug 3 consists of six cameras, five of which are mounted in a circular manner, one showing upwards, together covering 80 % of the full viewing sphere. Neighboring images only have a very small overlap, which is too weak for system calibration without additional information. We have mounted the omnidirectional multi-camera system Ladybug 3 on a robot, see Figure 3.17 (a), which executes a circular movement with a radius of 50 cm in a highly textured room while the Ladybug 3 is taking synchronized images. This ensures overlapping images of different cameras at different times of exposure. Initial values for this image sequence, consisting of 150 images taken by the five horizontal cameras at 30 different poses, are obtained with aurelo that provides 135,012 image points of 24,078 observed scene points. The resulting 150 camera poses are shown in Figure 3.17 (b).

After applying our bundle adjustment, the estimated a posteriori variance factor amounts to $\widehat{\sigma}_0^2 = 0.25^2$ using a priori stochastic model with $\sigma_l = 1$ pel for the image points, indicating the automatically extracted SIFT points to have a quite good precision. Two parallel walls with known distance of 7.01 m can be estimated out of the estimated scene points. We use the distance to define the scale between the estimated relative camera poses. The estimated rotation parameters show a very high precision and the maximum deviation to the manufacturer's calibration parameters amounts to 0.6°. The

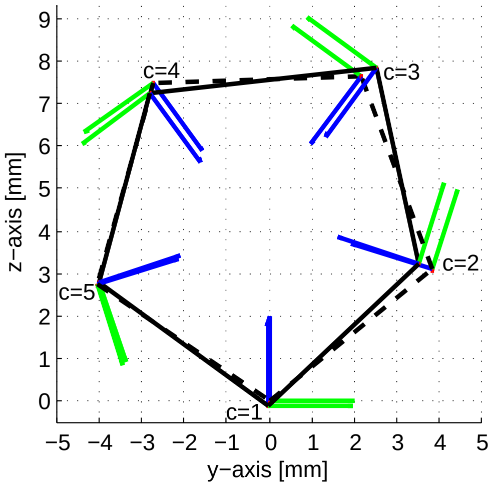


Figure 3.18: Comparison of relative poses: estimated (solid) and manufacturer given (dashed).

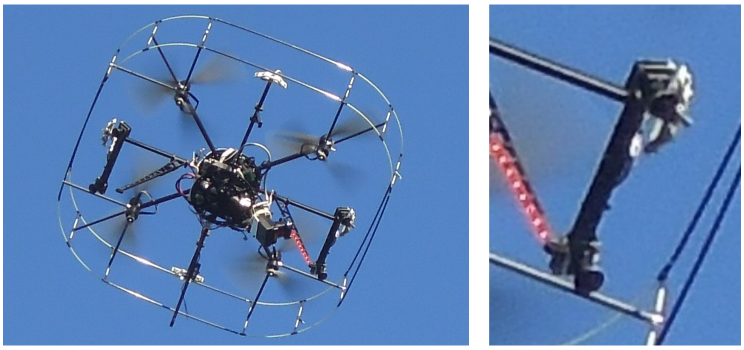
estimated precision of the rotations and translations between the cameras are in the order of 0.0015–0.0025° and 0.1–0.2 mm, respectively.

To compare the estimated poses with the ones provided by the manufacturer, we apply a rigid transformation, which minimizes the distances between the estimated and given projection centers. The resulting estimated relative poses in Figure 3.18 show translational deviations in the order of 1–4 mm compared to the manufacturer's calibration parameters. The reason for these deviations remains unclear.

The interior angles differ from a regular pentagon, where each interior angle is 108°, by up to 13°. Possible reasons for the deviations are too few observed scene points near the camera system and that we used an interior orientation for each camera from our own calibration, which is different from that of the manufacturer.

Multi-camera system with fisheye lenses. We make a similar investigation on an image sequence consisting of 96 images taken by four synchronized cameras with Lensagon BF2M15520 fisheye lenses having a field angle up to 185°. As described in Sec. 1.2, the cameras are mounted on an UAV to generate two stereo pairs, one looking ahead and one looking backwards, providing a large field of view, see Figure 3.19. The UAV moves along a circle at a height of 5 m above a parking lot while rotating around its own axis, providing four overlapping images at each time of exposure.

In order to find corresponding points using the SIFT operator, we need to use a projection of overlapping images which is not too far from a conformal projection, i.e. one that preserves angles, because of the severe fisheye-specific distortions at the image boundaries as the SIFT operator is only translation, scale and rotation invariant. For this reason, we transfer the original images into images following the stereographic fisheye model. This ensures a conformal mapping between two different images when observing a scene at infinity as they themselves are conformal mappings of the spherical image of the scene. We obtain



(a) UAV used in the MoD-project (b) Stereo camera

Figure 3.19: Illustration of the UAV. One stereo pair of the UAV is looking forward and one backwards, which provides a wide field of view.

low deviations from a similarity transformation for locally planar points not too close to the cameras, fulfilling the preconditions for rotation and scale invariant SIFT-matching.

aurelo provides approximate values for the 96 camera poses and 81,821 image points of 15,344 observed scene points. The image points are transformed into camera directions using Eq. (3.18). After bundle adjustment the estimated variance factor amounts to $\widehat{\sigma}_0^2 = 1.47^2$ using an a priori stochastic model with $\sigma_l = 1\,\mathrm{pel}$ for the image points, indicating a quite poor precision of the point detection. The cause for this low precision, which still needs to be analyzed, may be lower image quality caused by both, the fisheye projection and vibrations. The uncertainty of the estimated rotations and translations between the cameras within a stereo pair amounts to 2–6 mdeg and 0.5–1.5 mm, respectively, and the uncertainty of the estimated rotations and translations between the forward and backward looking stereo camera systems amounts to 5–9 mdeg and 1.5–2.5 mm.

3.8 Conclusion

In this chapter, we introduced a rigorous bundle adjustment for omnidirectional and multi-view cameras which enables an efficient maximum likelihood estimation with scene points being far or at infinity, which classical bundle adjustments are not capable of. Our estimation procedure is tailored to include points at infinity by using the homogeneous representation of scene points with spherically normalized coordinate vectors. The parameter estimation as well as the adjustment of the observations is applied in tangent space using the framework of reduced coordinates as proposed by (Förstner and Wrobel, 2016, Chap. 10). This way, a statistically rigorous estimation can be performed with minimal representations of homogeneous coordinates for image and scene points.

The evaluation of the simulated scenario demonstrates the correctness of the approach and the implementation. Feasibility on real data is demonstrated on a sequence of stereo images. The decrease of rotational precision when excluding far points is demonstrated on a dataset recorded by a multi-camera system and a single-view camera. The conducted experiments prove that far and points at infinity have a highly relevant influence on the rotational precision of the camera poses, especially if such points can be observed over long periods of time as in the image sequence recorded with the multi-camera system. The impact in the dataset of the single-view camera is not very high, as scene points with small intersection angles could only be observed for short periods of time in the image sequence.

Additionally, the bundle adjustment can be used to estimate the system calibration of a multi-camera system given the intrinsic calibration of the single-view cameras. No calibration targets are needed, just a movement of the multi-camera system taking synchronized images of a highly textured and static scene. As illustrated, multi-camera systems with non-overlapping views have to be rotated within the scene so that corresponding points are visible in different cameras at different times of exposure.

Experiments demonstrate the achieved precisions of the calibration of different multi-camera systems as a system with highly overlapping views, the omnidirectional multi-camera system Ladybug 3 and an omnidirectional camera system with fisheye lenses having a wide field of view.

4 Visual Odometry for Omnidirectional Camera Systems

In the previous chapter, we introduced our approach to bundle adjustment which allows to employ omnidirectional multi-camera systems and points at infinity. So far, we assumed to orient an image dataset after the acquisition of all images in a batch bundle adjustment. In this chapter, we introduce our online visual odometry system for pose estimation and sparse mapping from an image sequence. Our system employs an incremental version of the bundle adjustment introduced in the previous chapter to incrementally estimate the pose and map employing tracked visual features and optionally GPS and IMU information on keyframes. Fast pose estimation on frame rate is realized by robust resection on the incrementally refined map of 3D point coordinates.

The contribution of this chapter is a highly integrated system for fast and effective pose estimation and mapping on light-weight UAVs. Our approach provides an effective pose estimation, running at 10 Hz, that is computed fully on the copter using image data from an omnidirectional multi-fisheye camera system. The SLAM procedure combines spatial resection which is computed based on the map that is incrementally refined through bundle adjustment. The incremental bundle adjustment optionally combines the image data with raw real-time kinematic GPS (RTK-GPS) observations and inertial measurement unit (IMU) data on keyframes. In contrast to most existing systems, we fuse the image data with measured GPS carrier phase ranges, which allows us to exploit measurements in underconstrained situations, i.e. if only two or three satellites are visible. The estimation is done in a statistically sound manner and provides accurate 6 DoF pose estimates of the platform as well as accurate 3D locations of the feature points.

4.1 Introduction

Online pose estimation and mapping in unknown environments is essential for most mobile robots. Maps are needed for a wide range of applications and most robotic navigation systems rely on maps. Building such maps is often referred to as SLAM or simultaneous localization and mapping and a large number of different techniques to tackle this problem have been proposed in the robotics community. Popular filtering approaches rely on Kalman filters or particle filters and to emphasize their incremental nature, such filtering approaches are usually referred to as online SLAM methods. In contrast to that, most optimization approaches estimate the full trajectory and not only the current pose. They address the full SLAM problem and typically rely on least-squares or related optimization techniques.

We have implemented our pose estimation system running at 10 Hz as ROS modules. A specialized hardware based on a FPGA unit for state estimation, running at 100 Hz, provides GPS and IMU data. We employ a quadcopter, see Figure 1.2 on page 14, which is equipped with a GPS unit, an IMU and two fisheye stereo cameras for online pose estimation. The specialized hardware provides pose estimates at 100 Hz and can be used to control the copter.

4.2 Related Work

Simultaneous pose estimation and mapping has always been a central research focus in mobile robots, independently of the type of robot. This includes wheeled robots, underwater systems, or unmanned aerial vehicles. Thus, a large number of SLAM systems have been proposed.

After the work of Lu and Milios (1997), several systems have been proposed which address the full SLAM problem. A typical SLAM system, as the one presented in this thesis, is composed of a front-end and a back-end. The front-end acquires sensor measurements and performs data association to create observation equations. The back-end uses the equations to compute the parameters to make them maximally consistent with the observations. Like most structure from motion pipelines, as e.g. by Agarwal et al. (2011), recent visual SLAM systems employ bundle adjustment as the back-end for a final refinement. Efficient nonlinear optimization software packages, like g2o (Kümmerle et al., 2011), GTSAM (Dellaert, 2012) or Ceres solver (Agarwal et al., 2018), have been developed, which can be subsumed as graph-based optimization frame works. Such frame works simplify the formulation of complex optimization problems, they are independent of the sensing modality, and they are able to efficiently solve non-linear least squares optimization problems while exploiting sparsity.

There are several *approximate optimization techniques* approaches to ease computations and thus to obtain real-time capabilities for large scale environments. For example Olson et al. (2006) formulate SLAM as a pose graph optimization problem, which integrates transformation differences between camera poses for example obtained from odometry or loop closures. Indelman and Dellaert (2015) formulate bundle adjustment without 3D structure estimation by employing trifocal and epipolar constraints, which can be optimized rigorously with a Gauss-Helmert model or approximately but computationally more efficient with the afore mentioned factor graph based optimization frame works. Schneider et al. (2017) investigate the loss of precision induced by this approximation, which may be acceptable depending on the precision of the image point observations. Mouragnon et al. (2009), Engels et al. (2006) and Klein and Murray (2007) propose to perform local bundle adjustments to optimize over a sliding window containing the last recent images. This popular approach significantly reduces computational complexity but leads to global inconsistencies compared to full bundle adjustment. Other basic techniques for achieving real-time capabilities are based on Kalman filtering, e.g. by Davison (2003) or Choi and Lee (2012), which however are known to be inconsistent when applied to the inherently nonlinear SLAM problem, see Julier and Uhlmann (2001).

Also *globally optimal filtering techniques* based on bundle adjustment have intensively been investigated. They use current image information to improve the past pose and map information. Strasdat et al. (2012) show that filtering all frames is inferior to using keyframes and that a high number of features is superior to a high number of frames. Incrementally updating the normal equations can be replaced by updating the QR-factorization, described in detail in (Golub and Loan, 1996) and e.g. proposed for aerial on-line triangulation (Grün, 1984). Real-time bundle adjustment has been tackled intensively in the photogrammetric community, see e.g. the review by Grün (1987). Kaess et al. (2012) realized a completely incremental nonlinear least squares estimation algorithm called iSAM2 which effectively re-uses the previously computed solution. The algorithm is implemented in the factor based estimation frame work GTSAM and allows for incremental relinearization and reordering by employing the probabilistic structure of the Bayes-tree, see Kaess et al. (2010). In this work, we apply the bundle adjustment proposed in Chap. 3 based on keyframes and employ iSAM2 to solve it incrementally in order to estimate a globally consistent solution on the UAV in real-time.

While most of the SLAM back-ends are independent from the sensing modality, several systems have been tailored to visual SLAM. In this context, *dense 3D reconstruction* approaches have been proposed such as DTAM by Newcombe et al. (2011) or the approach by Stühmer et al. (2010) which computes a dense reconstruction using variational methods. Optimizing the dense geometry and camera parameters is possible but a rather computationally intensive task, see Aubry et al. (2011). To tackle the computational complexity for real-time operation, semi-dense approaches have been proposed, for example the semi-dense direct approach LSD-SLAM by Engel et al. (2013) and the sparse direct method DSO-SLAM by Engel et al. (2018) which optimize a photometric error and do not rely on pre-computed keypoints. In this thesis, we use tracked keypoint features like in the popular ORB-SLAM approach, see Mur-Artal et al. (2015).

Due to the low weight of cameras, *visual reconstruction techniques for light-weight UAVs* received considerable attention. Pizzoli et al. (2014) propose a reconstruction approach for UAVs that combines Bayesian estimation and convex optimization. They execute the reconstruction on a GPU at frame rate. Also combinations of cameras on an indoor UAV and RGB-D cameras on a ground vehicle have been used for simultaneous localization and mapping tasks aligning the camera information with dense ground models, see Forster et al. (2013). Harmat et al. (2015) adapted the parallel tracking and mapping (PTAM) algorithm by Klein and Murray (2007) to handle omnidirectional multi-camera systems to estimate the pose of a small UAV. Onboard methods for autonomous navigation of an UAV exploiting a Kalman filter to process stereo camera and IMU input are presented by

Rudolf et al. (2010) and Tomic et al. (2012), in the latter additionally with laser input. Ellum (2004) investigates the accuracy and reliability of tight coupling of raw GPS code pseudo-ranges into an offline bundle adjustment. In this work, our online SLAM procedure combines the image data with GPS carrier phase observations and IMU data incrementally, such that we obtain a precise pose and feature map even in cases where the GPS is not observable or underconstrained.

Currently work has also been published that shows success in solving subtasks of the SLAM problem with deep neural networks. Kendall et al. (2017) use deep learning to learn an end-to-end mapping from an image pair to disparity maps and achieve good results on traditionally difficult scenes, e.g. with low texture or complex geometry or occlusions. Han et al. (2015) perform feature matching by training a patch matching system end-to-end and Kendall et al. (2015) train a convolutional neural network to obtain the 6 DoF camera pose from a single image.

4.3 Online Pose Estimation and Mapping

Visual odometry consists in determining the pose of the cameras in real-time. Our camera setup utilizes four cameras arranged as two stereo pairs on a quadcopter, see Figure **??**. One stereo camera looks ahead and the other one looks backwards, both tilted at an angle of 45°. Equipped with fisheye lenses with 185° field angle, the cameras cover a large area around the UAV at each time of exposure. The cameras are triggered synchronously at 10 Hz and the basis of the stereo cameras amounts to 20 cm.

We refer to images taken at the same time of exposure as a frame set. The pose determination of each frame set relies on image feature points with known association to scene points in an incrementally refined and extensible map. The estimation and refinement of the map is performed in a bundle adjustment on selected keyframes that also optionally integrates GPS as well as IMU data, running in parallel. The overall system is designed such that all processing can be done online on an onboard PC.

Our approach requires calibrated cameras. We calibrate the intrinsic parameters of each fisheye camera in advance according to Abraham and Hau (1997) as detailed Sec. 2.2.2. For calibration, we model the fisheye lens with the equidistant-model allowing for ray directions with an angular distance larger than 90° to the viewing direction, see Eq. (2.20). The mutual orientation of the fisheye cameras in the multi-camera system is determined in advance with a system self-calibrating bundle adjustment, as described in the previous section. We further observe GPS control points in the images to derive the offset of the camera-system to the phase center of the GPS antenna in advance.

4.3.1 Overview

The overall process consists of the following steps:

1. The data acquisition and association detects feature points and performs the match-

ing to provide corresponding image points to the previous frame set and the other cameras.

2. The orientation of each frame set with resection provides a fast pose estimate and allows to select keyframes.
3. An incremental bundle adjustment merges the new information at a keyframe with the previous information in a statistically optimal way.

We aim at efficient methods for reliable data association, for fast pose determination, and target an outlier-free information for the bundle adjustment step. This optimization step is the most costly one as it uses all available data on the selected keyframes. To avoid long computation times, the optimization is performed with the incremental optimization iSAM2 (incremental smoothing and mapping) by Kaess et al. (2012). The remainder of this section describes the three steps in detail.

4.3.2 Visual Data Acquisition and Association

Our visual pose estimation and mapping procedure exploits point features extracted from the images. To allow for handling four cameras onboard the copter, an efficient feature extractor is essential. To this end, we select KLT features that are tracked in the individual cameras. We detect interest points that are corners in the gradient image with a large smallest eigenvalue of the structure tensor, see Shi and Tomasi (1994), and track them with the iterative Lucas-Kanade method with pyramids according to Bouguet (2000). Figure 4.1 shows an example of tracked interest points in the four fisheye images of a frame set.

Having calibrated cameras, each tracked feature point can be converted into a ray direction $\mathbf{x}'$ that points in the individual camera system to the observed scene point. Additionally, we transform the uncertainty of the image coordinates to the uncertainty of $\mathbf{x}'$ via variance propagation yielding $\Sigma_{\mathrm{x}'\mathrm{x}'}$ as described in Sec. 3.5.2. In all cases, the covariance matrix of the camera rays is singular, as the normalized 3-vector only depends on two observed image coordinates. We use the camera rays with its covariance information for the spatial resection at frame rate and for incremental bundle adjustment on keyframes. Both methods will be detailed in the following two sections.

To match feature points between the stereo camera pairs, we determine correlation coefficients between image patches at the feature points in the left and right images. We exploit epipolar geometry to reduce candidates within the propagated error bounds of the corresponding epipolar lines, see Figure 4.2. We assume feature points with the highest correlation coefficient ρ_1 to match if (a) ρ_1 is above an absolute threshold, e.g. 0.8, and if (b) – if there is more than one candidate close to the epipolar line – the closest-to-second-closest-ratio $r = \rho_2/\rho_1$ with the second highest correlation coefficient ρ_2 is lower than an absolute threshold, e.g. 0.7. Finally we check if this criterion holds also for all feature points in the left image if there are more than one feature points on the corresponding epipolar lines. In some rare cases this procedure leads to wrong matches, which can be

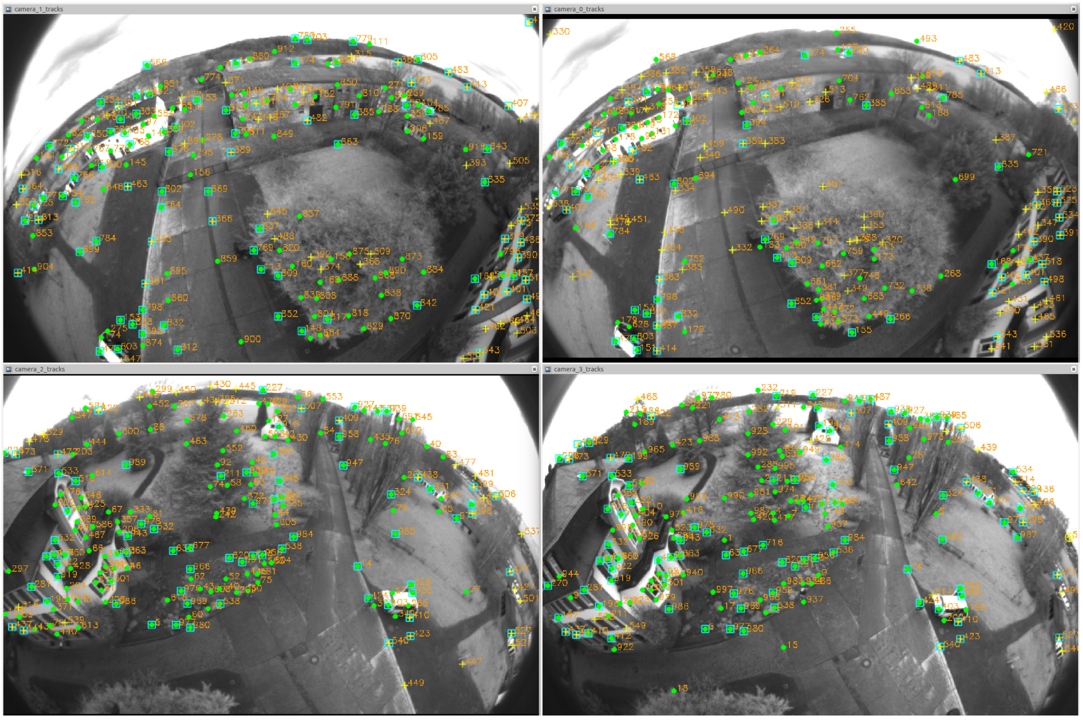


Figure 4.1: Synchronized triggered frame set of the four fisheye cameras. Each image contains around 200 feature points that are tracked using a KLT tracker. The top row shows the stereo image pair of the ahead looking stereo camera, the bottom row of the backwards looking stereo camera.

detected with a third observing ray from another pose.

4.3.3 Fast Pose Estimation

In our approach, we use feature maps, which are defined as a set of scene points $\mathcal{X} = \{\mathcal{X}_i\}$. In theory, the location of these scene points and the pose of the camera system can be estimated through bundle adjustment directly. Given the computational demands, it is impossible to compute a bundle adjustment solution at 10 Hz on the copter. Therefore, we execute the bundle adjustment only on selected keyframes at around 1 Hz. To compute the camera poses between the keyframes, we compute the UAV poses by spatial resection on each frame set.

The location of the points is initialized at the first acquired frame set by forward intersecting the matched ray directions in the stereo pairs. The frame set that initializes $\mathcal{X}$ is chosen as first keyframe $\mathcal{K}_1$.

After initialization of the map, the motion M_t of the camera system in relation to the map is computed at frame rate using resection. For resection we use scene points $\mathbf{X}_i$ that are observed in cameras $c = 1, ..., 4$ at time t and exploit the known system calibration M_c to consider the multiple projection centers. Each M_c describes the known transformation of a single camera c to the reference frame of the UAV and M_t describes the unknown transformation of the UAV reference frame into the reference coordinate system of the

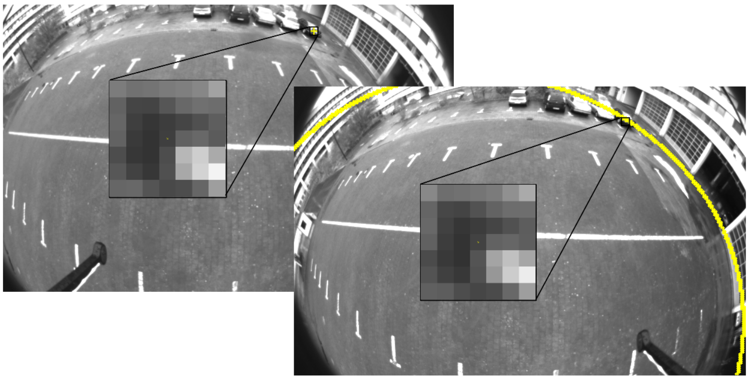


Figure 4.2: Example images taken in a synchronized way in the left and right camera of a stereo pair. The extracted feature point in the left image on the rightmost car has the illustrated epipolar line in the right image. The matching point in the right image lies on the indicated yellow epipolar line and the corresponding local image patches show a high correlation. Note that epipolar lines in fisheye images are curves and not straight lines as in perspective images.

map at time t, thus M_t contains the pose parameters of the UAV. Following Eq. (3.47), an estimated pose $\widehat{\mathsf{M}}_t^a$ induces the reduced 2D residual vector $\widehat{\boldsymbol{v}}^a_{x_{r,itc}}$

$$\boldsymbol{x}_{r,itc} + \widehat{\boldsymbol{v}}^a_{x_{r,itc}} = \operatorname{null}(\widehat{\mathbf{x}}^{a\mathsf{T}}_{itc})^{\mathsf{T}} \mathsf{N}\Big(\mathsf{P}_c \, \widehat{\mathsf{M}}_t^{a^{-1}} \mathbf{X}_i\Big) \tag{4.1}$$

between the reduced 2D coordinates $\boldsymbol{x}_{r,itc}$ of an observed ray direction $\mathbf{x}_{itc}$ and the reduced 2D coordinates of a predicted ray direction $\widehat{\mathbf{x}}^a_{itc} = \mathsf{N}(\mathsf{P}_c \, \widehat{\mathsf{M}}_t^{a^{-1}} \mathbf{X}_i)$, which points to homogeneous scene point $\mathbf{X}_i$ in camera c at time t. The homogeneous scene point $\mathbf{X}_i$ is transformed with the inverse of the estimated motion $\widehat{\mathsf{M}}_t^a$ and projected with system calibration $\mathsf{P}_c = [I_3 \,|\, \mathbf{0}_3] \, \mathsf{M}_c^{-1}$ into a single camera into the predicted direction, which is spherically normalized to unit length with $\mathsf{N}(\mathbf{x}) = \mathbf{x}/|\mathbf{x}|$.

We optimize the six pose parameters of M_t^a with an iterative maximum likelihood-type estimation with the robust Huber cost function (Huber, 1981) that down weights observations with large residuals, as discussed in Sec. 2.3.2. The estimation of the pose parameters for $\widehat{\mathsf{M}}_t$ converges in 2 - 3 iterations using the estimated pose of the previous frame set $\widehat{\mathsf{M}}_{t-1}$ as initial value. This allows a robust pose estimation at a high frame rate.

To obtain a near outlier-free input for bundle adjustment, we exploit the estimated weight in the Huber cost function. Observations with low weights are considered as outliers and are not used in bundle adjustment and excluded from tracking. Image points which are excluded or could not be tracked into the current frame are replaced by new interest points.

4.3.4 Keyframe-Based Incremental Bundle Adjustment

The last step in our visual odometry pipeline is keyframe-based bundle adjustment, which reduces the processing to some geometrically useful, tracked observations. This optimiza-

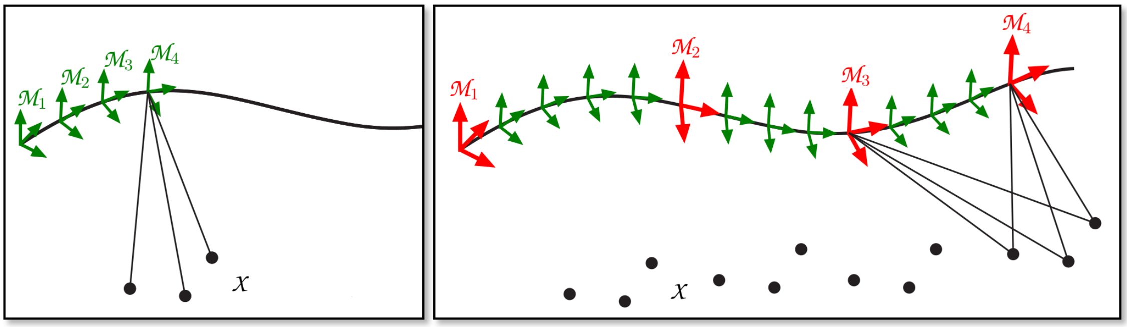


Figure 4.3: Illustration of the keyframe-based bundle adjustment. Left: at every frame set, a new motion state $\mathcal{M}_t$ (green) is calculated by spatial resection using the scene points in the map $\mathcal{X}$. Right: after a certain motion distance, e.g. 1 m or 30°, a keyframe $\mathcal{K}_t$ is initiated (red). At every keyframe, a fast incremental bundle adjustment step is calculated to refine all keyframe poses $\mathcal{M}_t \in \mathcal{K}$ and scene points in $\mathcal{X}$.

tion step considers the information from the camera images. It also allows to incorporate GPS observations as well as IMU measurements, which is addressed in the next section.

For our real-time applications the processing of a keyframe needs to be finished by the time the next keyframe is added. For the optimization, we use iSAM2 (Kaess et al., 2012), which models the problem as a factor graph and allows for very efficient incremental nonlinear optimization that reuses information obtained from optimizing camera poses and scene points in the previous time steps, as introduced in Sec. 2.4. Each node on the factor graph corresponds to a keyframe pose $\mathcal{M}_t$ or a 3D scene point $\mathcal{X}_i$. The nodes are connected through factors that result from the different observations. We define the update rules for estimated corrections for the pose parameters according to Eq. (3.30) and for the scene points according to Eq. (3.45).

We add a new keyframe $\mathcal{K}_t$ in case a certain geometric distance to the last keyframe $\mathcal{K}_{t-1}$ is exceeded, see Figure 4.3. Each new keyframe contains two kinds of observations, χ_1 and χ_2, where χ_1 are the observations of scene points that are already in the map and χ_2 denotes those observing new scene points. With each new keyframe the map is expanded by forward intersection with observations χ_2. Note that only χ_1 has been revised from outliers in the robust pose estimation described previously. In order to identify outliers in χ_2 based on their residuals, we require a track to consist of at least three keyframes for mapping.

The map $\mathcal{X}$ and keyframe poses in $\mathcal{K}$ are simultaneously refined using bundle adjustment. In terms of factor-graphs each observed camera ray $\mathbf{x}'_{itc}$ produces a factor $\phi_{itc}(\mathcal{M}_t, \mathcal{X}_i; \chi_{itc})$, see Figure 4.4. We define for each factor $\phi_{itc}(\mathcal{M}_t, \mathcal{X}_i; \chi_{itc})$ the measurement equations according to Eq. (3.47) and specify the linearization w.r.t. the scene point and pose parameters according to Eq. (3.55) and Eq. (3.56), respectively. The reduced covariance matrix in Eq. (3.40) specifies the stochastic model for the observations of each factor.

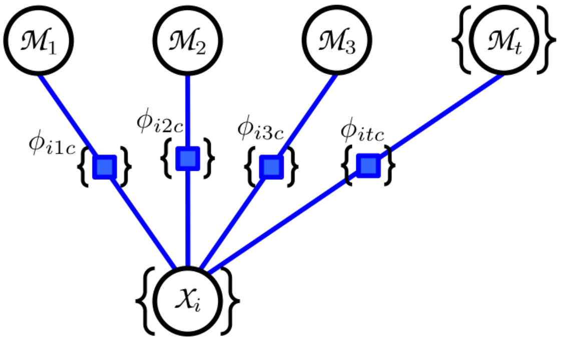


Figure 4.4: Illustration of the factor graph of the keyframe-based bundle adjustment described in Sec. 4.3.4 with camera information only. A scene point $\mathcal{X}_i$ and a keyframe pose $\mathcal{M}_t$ are connected by a factor node $\phi_{itc}(\mathcal{M}_t, \mathcal{X}_i)$ if at least one camera c of the multi-camera system observes scene point i at pose t.

For the first ten keyframes we use batch bundle adjustments as the map contains only a small number of scene points yet. After that, the new information is incrementally merged with the previous information, yielding a fast optimal solution for the bundle adjustment using iSAM2. As new measurements often have only a local effect and fill-in may become expensive, iSAM2 encodes the conditional density of cliques in a Bayes tree, which allows for an efficient incremental reordering, just-in-time relinearization and partial solving, if parameters change only locally. For more details, please refer to (Kaess et al., 2012).

4.3.5 Integration of GPS and IMU Information

So far, our SLAM procedure combines spatial resection computed based on the map that is incrementally refined through bundle adjustment. In the following, we optionally incorporate GPS carrier phase measurements and IMU data on keyframes.

Whenever UAVs operate outdoors, they typically make use of GPS observations for global positioning. Usually, GPS-based state estimation on light-weight UAVs is based on a L1 C/A-code GPS receiver, MEMS inertial sensors and a magnetometer, see e.g. Yoo and Ahn (2003), Kingston and Beard (2004) or Wendel et al. (2006). Such a sensor combination only leads to global position accuracies of approx. $2-10\,\text{m}$ and attitude accuracies of approx. $1-5°$. This is often good enough to autonomously follow waypoints, but it is typically insufficient for UAV control or for geodetic-grade surveying and mapping applications. First systems that realized cm-accurate real-time kinematic GPS (RTK-GPS) solutions on UAVs, were presented by Rieke et al. (2011), Stempfhuber and Buchholz (2011), Bäumker et al. (2013) and Rehak et al. (2014). In none of these developments, however, the position and attitude estimation is performed in real-time onboard of the UAV platform, which especially for the UAV control and precise autonomous flight is key to robust operation.

In order to build a highly integrated online SLAM system – as the one we propose here – it is important to have access to all raw measurements as well as the state estimation algo-

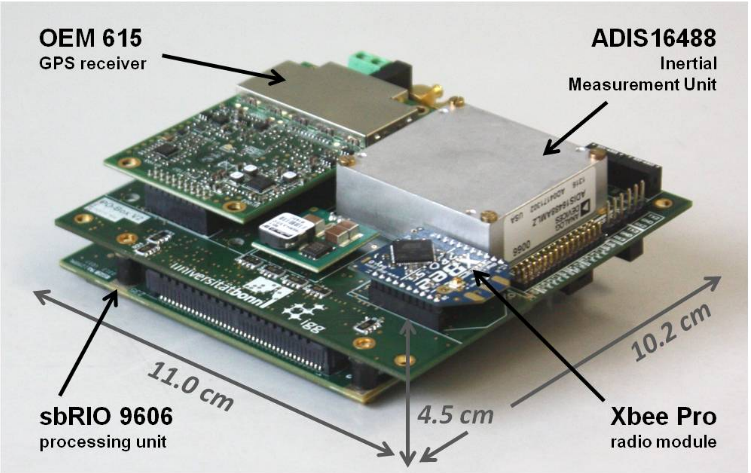


Figure 4.5: The RTK-GPS/IMU state estimation board based on a 400 MHz processor and an FPGA. The setup includes a geodetic-grade GPS receiver (Novatel OEM 615), a low-cost single-frequency GPS Chip (Ublox LEA6T), an IMU (Analog devices ADIS16488) and a magnetometer (Honeywell HMC5883L). Image from Eling et al. (2015).

rithms. Therefore, we employ the state estimation board shown in Figure 4.5, developed by Eling et al. (2015), which gives us full control over the measurements, algorithms, and internal states. This enables us to effectively incorporate camera information with GPS carrier phase measurements and inertial sensor readings in our real-time SLAM system on the UAV on the level of raw observations. Additionally, our SLAM solution is able to exploit information from underconstrained RTK-GPS situations, i.e., less than four available satellites when standard GPS receivers report a GPS loss and cannot estimate a solution, due to the integration of raw carrier phase ranges to handle underconstrained RTK-GPS situations effectively.

GPS positioning of mobile objects based on carrier phase ranges in real-time is called RTK-GPS and it is a relative positioning procedure, in which the unknown coordinates of a movable station are determined with respect to a stationary master station. The advantage of this relative positioning is an improved accuracy that comes from single- and double-differencing of the observations, see Hofmann-Wellenhof et al. (2008, Chap. 6.2). By using single-differences, which are calculated from a signal of one satellite measured at both receivers (UAV and the master), the satellite clock bias as well as the atmosphere refractions can be reduced significantly. Double-differences are calculated from the single-differences of two satellites and therefore eliminate the receiver clock bias and other receiver dependent effects.

The mathematical model of a double-difference (DD) carrier phase observation $\Phi_{RM}^{S_j S_0}(t)$, as it is used in the state estimation board and in the SLAM system, is

$$\Phi_{RM}^{S_j S_0}(t) = \frac{1}{\lambda}\rho_{RM}^{S_j S_0}(t) + \left(N_R^{S_j S_0} - N_M^{S_j S_0}\right) + \epsilon_{RM}^{S_j S_0}, \tag{4.2}$$

with the DD phase measurement $\rho_{RM}^{S_jS_0}(t)$ from satellites S_j and S_0 at robot R (UAV) and master M at time t, expressed in cycles, and signal wavelength λ and DD geometric range $\rho_{RM}^{S_jS_0}(t)$. The term $\epsilon_{RM}^{S_jS_0}$ denotes the measurement noise and the integer values $N_R^{S_jS_0}$ and $N_M^{S_jS_0}$ are time independent single-difference ambiguities. To simplify matters, the terms of remaining systematic errors in Eq. (4.2) are neglected.

As the receiver is only able to measure the fractional part of a carrier wave cycle, the remaining integer number of cycles needs to be resolved. Eling et al. (2015) estimate this number as real valued ambiguities within the GPS/IMU integration using the Kalman filter and fix the ambiguities to integer numbers by applying the modified LAMBDA method proposed by Chang et al. (2005). Due to the GPS/IMU integration, cycle slips in the carrier phases can be detected and repaired reliably, see Eling et al. (2014) for further details.

GPS information is usually integrated as preprocessed 3D positions. But in situations in which less than 4 satellites are available standard GPS receivers report a GPS loss and cannot estimate and provide a solution. Through the combination of visual SLAM and DD measurements, we can however compute a solution and exploit individual double differences.

In the following, we define the body frame of the sensor system to coincide with the antenna's phase center and to be aligned to the axes of the IMU. We observe GPS control points in the images of the single-view cameras with known system calibration to derive the offset of the multi-camera system to the body frame. This way, each M_c can be transformed to describe a motion from the antenna's phase center to each single view camera.

To integrate the GPS double differences, the keyframe poses M_t need to be in the GPS coordinate system. Therefore, we initially require to have a unique GPS solution, for which at least three double differences are needed. When initializing the bundle adjustment, we first determine the positions of the first five keyframes with GPS coordinates and do not integrate double differences into the bundle adjustment. From the 5th keyframe with a GPS position on, we estimate a similarity transformation and transform all keyframe poses and the map into the GPS system.

Then, we integrate the DD carrier phase observations by adding a factor $\phi_t(\mathcal{M}_t)$ for the L1 and L2 frequency to the factor graph, see Figure 4.6. For the measurement equation, the coordinates of the GPS satellites and the master receiver are needed, see Hofmann-Wellenhof et al. (2008). The position $\boldsymbol{Z}_t$ of the movable receiver R needs to be estimated. An estimated position $\widehat{\boldsymbol{Z}}_t^a$ of the UAV, initially given e.g. from resection, induces the residual $\widehat{v}^a_{\Phi_{RM}^{S_jS_0}}$ in each DD measurement equation

$$\Phi_{RM}^{S_jS_0} + \widehat{v}^a_{\Phi_{RM}^{S_jS_0}} = (d_M^{S_0} - d_R^{S_0}(\widehat{\boldsymbol{Z}}_t^a)) - (d_M^{S_j} - d_R^{S_j}(\widehat{\boldsymbol{Z}}_t^a)) \tag{4.3}$$

where $d_M^{S_0}$, $d_R^{S_0}$, $d_M^{S_j}$ and $d_R^{S_j}$ are distances between the receivers and satellites as illustrated

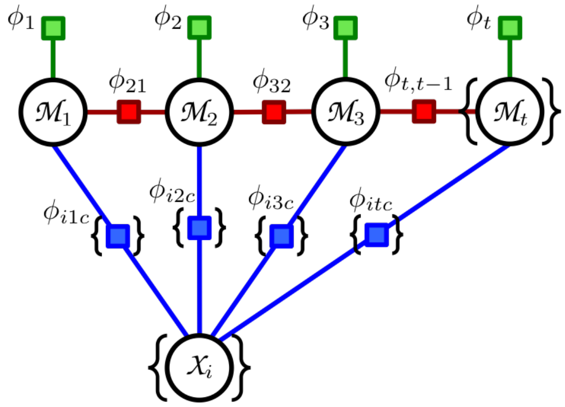

ϕ_1
ϕ_2
ϕ_3
ϕ_t
ϕ_{21}
ϕ_{32}
$\phi_{t,t-1}$
$\mathcal{M}_1$
$\mathcal{M}_2$
$\mathcal{M}_3$
$\mathcal{M}_t$
ϕ_{i1c}
ϕ_{i2c}
ϕ_{i3c}
ϕ_{itc}
$\mathcal{X}_i$

Figure 4.6: Illustration of the factor graph of the keyframe-based bundle adjustment which integrates GPS and IMU information in addition to the camera information. In addition to the factor nodes $\phi_{itc}(\mathcal{M}_t, \mathcal{X}_i)$ depicted in Figure 4.4, we have factor nodes $\phi_t(\mathcal{M}_t)$ which integrate the DD carrier phase observations on keyframe t and factor nodes $\phi_{t,t-1}(\mathcal{M}_{t-1}, \mathcal{M}_t)$ which integrate the measured rotation difference between to succeeding keyframes $t-1$ and t observed by the IMU.

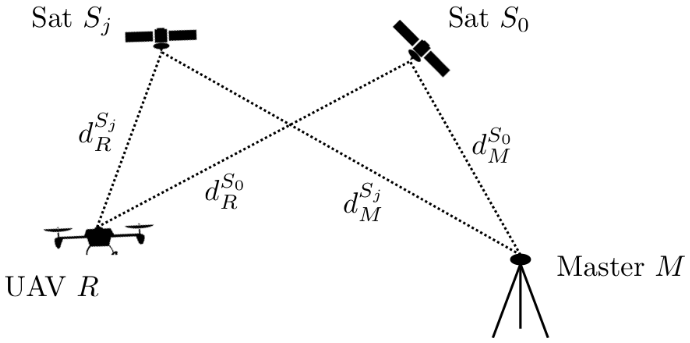

Sat S_j
Sat S_0
$d_R^{S_j}$
$d_M^{S_0}$
$d_R^{S_0}$
$d_M^{S_j}$
UAV R
Master M

Figure 4.7: Double differences are determined using the distances d between the known positions of the GPS satellites and the master and the approximate UAV position.

in Figure 4.7.

Micro-Electro-Mechanical System (MEMS) based IMU outputs are corrupted by significant sensor errors due to the integration of acceleration, forces and angular velocities. The measured IMU rotation angles are integrated over time between two neighboring keyframes leading to the observed angles in the 3-vector $\boldsymbol{r}_t$ with the rotation matrix $R(\boldsymbol{r}_t)$. The measurement equation reads as

$$R(\boldsymbol{r}_t + \widehat{\boldsymbol{v}}^a_{r_t}) = \widehat{R}^{a\mathsf{T}}_{t-1}\widehat{R}^a_t \tag{4.4}$$

with residuals $\boldsymbol{v}_{r_t}$ and is integrated into the factor graph with factor $\phi_{t,t-1}(\mathcal{M}_{t-1}, \mathcal{M}_t)$, see Figure 4.6.

4.4 Experiments

The approach described above has been implemented in ROS and runs on the onboard PC of the UAV. The experimental evaluation is designed to illustrate the performance of the approach. In its first section, experiments demonstrate the potential of the incremental bundle adjustment w.r.t. time requirements, real-time capabilities and optimality. Experiments of the second section investigate the localization accuracy of visual odometry with integrated GPS information. The experimental evaluation of section three illustrates the advantage of incorporating incomplete GPS observations with less than four satellites on the level of carrier phase observations as well as the potential of the overall system for highly accurate and georeferenced pose and map estimation.

4.4.1 Real-time Capabilities and Optimality of Incremental Bundle Adjustment

To test the real-time capabilities and the optimality of the incremental bundle adjustment, we investigate the required time to incrementally process a keyframe and its dependency on the number of new factors and number of affected parameters. Subsequently, we examine the accuracy of the tracked feature points, and the optimality of the incremental bundle adjustment by comparing its results to a batch bundle adjustment.

For our investigations, we employ an image sequence taken with the four fisheye cameras from our UAV performing two circular motions. The image sequence consists of 1,800 frame sets taken with 14 Hz. We apply a high-weighted prior on the 6D pose of the first keyframe to define the coordinate system of the map. The scale is defined by the known mutual orientations in the multi-camera system.

The choice of relinearization threshold β of the iSAM2 algorithm has a significant influence on the required time and the obtained accuracy of the estimated parameters. Setting the threshold β for linearization too low leads to relinearization of all variables on every keyframe and setting β too large decreases the accuracy of the estimates.

Our system initiates a new keyframe after each 1 m resulting in 107 keyframes. Tracking 50 feature points in each camera and setting β for the rotations to 0.5° and for the translations to 3 cm yields a very fast processing of the bundle adjustment that is always faster than 1 second on a 3.6 GHz machine, see Figure 4.8 (a). The required time is independent of the number of new factors added to the Bayes tree, see Figure 4.8 (c), but rather highly depends on the number of cliques related to variables that need to be relinearized, see Figure 4.8 (b).

The root mean square error (RMSE), which is determined after each incremental bundle adjustment, is in the order of 2 - 3 pixel, see Figure 4.9. This is quite large as we assumed a standard deviation of $\sigma_l = 1$ pixel for the extracted feature points.

To check the optimality of the incremental bundle adjustment and to examine the accuracy of the image features, we apply the batch bundle adjustment BACS on the observa-

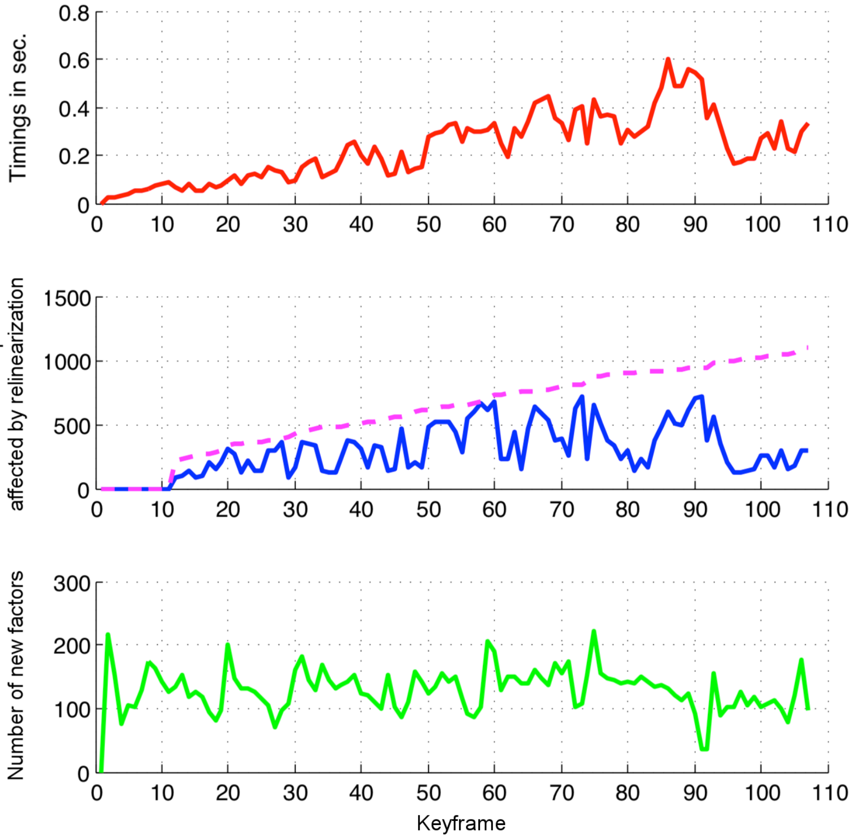


Figure 4.8: Illustration of the required time to incrementally process a keyframe and its dependency on the number of new factors and the number of affected parameters. (a) Required time for processing incremental bundle adjustment using iSAM2. (b) Number of cliques related to relinearized variables (solid) and the total number of cliques in the Bayes tree (dashed), note the effect on (a). (c) Number of new factors added, note that the number has no effect on (a).

tions used for the incremental bundle adjustment with iSAM2. We use the incrementally estimated values as approximates and retain the pose of the first keyframe to use the same gauge definition as we did using iSAM2. Using an a priori precision of $\sigma_x = 1$ pixel, we obtain an estimated variance factor of $\widehat{\sigma}_0^2 = 2.0^2$ which is in the order of the RMSE. Applying the robust Huber minimizer shows no significant outliers and yields an equal robust estimated variance factor of $\widehat{\sigma}_0^2 = 1.96^2$. The image point precision of 2 pixel may be seen as somewhat low, but results from the frame-wise KLT tracking.

Differences in the estimated pose parameters between those from the incremental bundle adjustment using iSAM2 and the batch bundle adjustment using BACS are shown in Figure 4.10 for each set of keyframes. These differences are within their estimated uncertainty, which is up to 0.3° in rotations and up to 8 cm in translations. This shows that our threshold β, which is 0.5° for the rotations and 3 cm for the translations, appears to be reasonable. The deviations between the estimated rotations around the x-axis show a small continuous trend to fall below zero, which could be reduced by lowering the thresh-

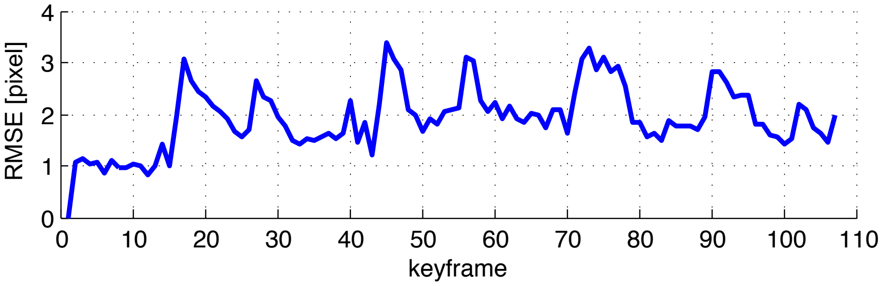


Figure 4.9: Root mean square error of extracted image points for each keyframe.

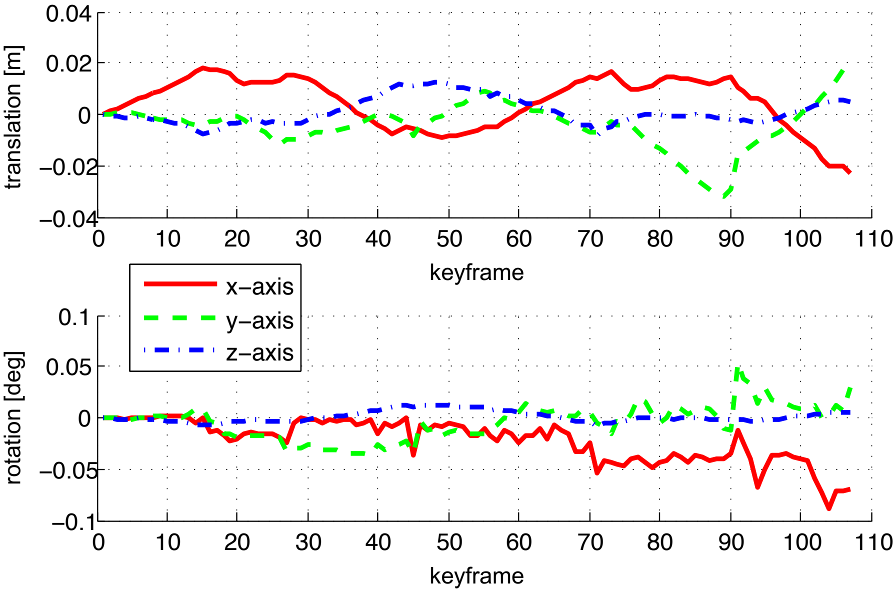


Figure 4.10: Deviations between the estimated rotation angles and translations of BACS and iSAM2 on all keyframes. The z-axis points in flight direction, the x-axis points upwards and the y-axis is orthogonal to both.

old β. Additionally, the results show that iSAM2 provides estimates which are optimal in a statistical sense, like the rigorous batch bundle adjustment BACS.

4.4.2 Localization Precision of Visual Odometry with Integrated GPS

In order to investigate the precision gain obtained by the integration of GPS information, we employ sensor data that was recorded by the UAV during a 5 min flight in which a building was mapped with the high resolution camera in a distance of about 5 m. In this flight, the visual odometry sets a new keyframe on average after 2 seconds. The processing of a new keyframe needs on average 0.3 to 0.5 seconds. In most cases this time is sufficient (1) to detect and track 200 feature points in each of the four cameras with a frame rate of 10 Hz, (2) to determine the spatial resections for each frame set, (3) to revise the tracks from outliers and (4) to execute the incremental bundle adjustment step.

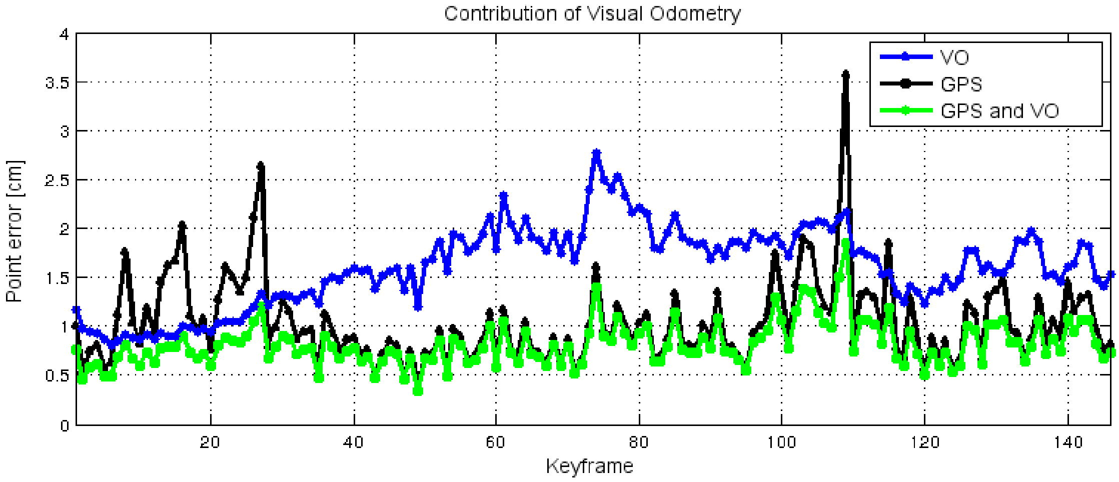


Figure 4.11: The precision of the positions from GPS (black), from pure visual odometry (blue) and from visual odometry which integrates GPS (green), respectively shown as point errors $\sqrt{\sigma_X^2 + \sigma_Y^2 + \sigma_Z^2}$. The theoretical precision of pure visual odometry is derived from the difference: Apparently the visual odometry has a standard deviation below 3 cm and on average is up to twice as uncertain as the GPS measurements, but for short time intervals it provides more precise positions. The uncertainty of the integrated position throughout is less than 2 cm.

The positions and their covariance information obtained with the direct georeferencing unit are integrated as uncertain prior information on the keyframe's positions to obtain long-term stability, georeferenced poses and increased accuracy and precision. The incremental bundle adjustment integrating all information can determine a real-time position with a less volatile standard deviation below 2 cm, see Figure 4.11.

To empirically validate the obtained theoretical a posteriori uncertainties, we determine the trajectory with pure visual odometry without using prior information from GPS. Using a similarity transformation on the GPS positions we can determine deviations between the independently estimated trajectories. The deviations between the keyframe poses are shown in the histograms in Figure 4.12. The histograms confirm the theoretical standard

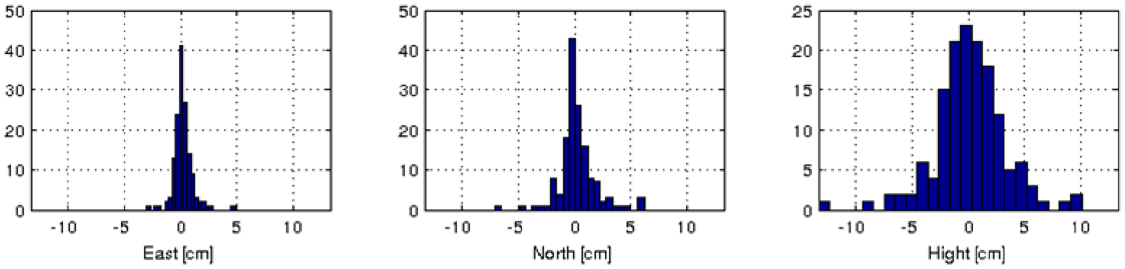


Figure 4.12: The deviations between the keyframe positions from visual odometry and the GPS coordinates.

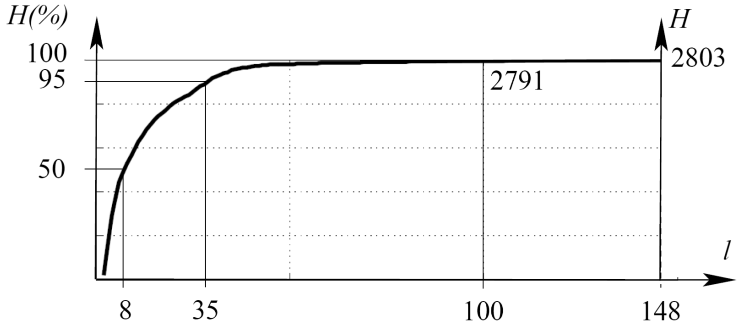


Figure 4.13: Cumulative histogram $H(l)$ of the track lengths l of a flight with four fisheye cameras. The median and the 95-%-point are indicated.

deviation from Figure 4.11.

In total, 2,803 scene points have been tracked. Figure 4.13 shows the cumulative percentage of the 2,803 track lengths. For this flight with 148 keyframes most track lengths contain eight keyframes, 5 % of the tracks contain 35 keyframes and twelve tracks contain at least 100 keyframes. As a consequence we obtain a high long-term stability for the orientation angles. The obtained real-time accuracy of the rotations throughout is in the order of about 0.05 - 0.1°. Especially scene points close to infinity, i.e. points that are far away relative to the motion of the observing camera system, can be observed for a long time, which increases the accuracy of the camera rotation as shown in the previous chapter.

4.4.3 Integration of GPS Carrier Phase Observations

Our experimental evaluation is designed to illustrate the accuracy of pose estimation for light-weight UAV integrating the visual information as well as GPS and IMU information. We now investigate the benefit of the GPS integration on the level of double difference observations, which is able to exploit incomplete GPS observations with less than 4 satellites. Additionally, we show that our system provides highly accurate and georeferenced pose and map estimation. For evaluation, we recorded all sensor data with our UAV under good GPS conditions, with 5 to 8 visible satellites. This allows us to manually eliminate GPS observations and evaluate the effect on the overall state estimation procedure. The flight used for this evaluation guided the UAV along the facade of a house, the variation in position is around 60 m and 15 m in height, see Figure 4.14. The dataset contains 3,368 frame sets recorded at 10 Hz and around 200 features are tracked in each of the four fisheye cameras. The SLAM system initiates a keyframe every 1 m, resulting in 274 keyframes and online SLAM starts at take off on the ground and ends at the landing.

The first experiment is designed to show the obtained accuracy of the estimated keyframe poses during the whole flight. To obtain the theoretical accuracy, we extract the covariance information when the bundle adjustment is incrementally solved at a new keyframe. This is too time consuming for online processing but can be done as an offline

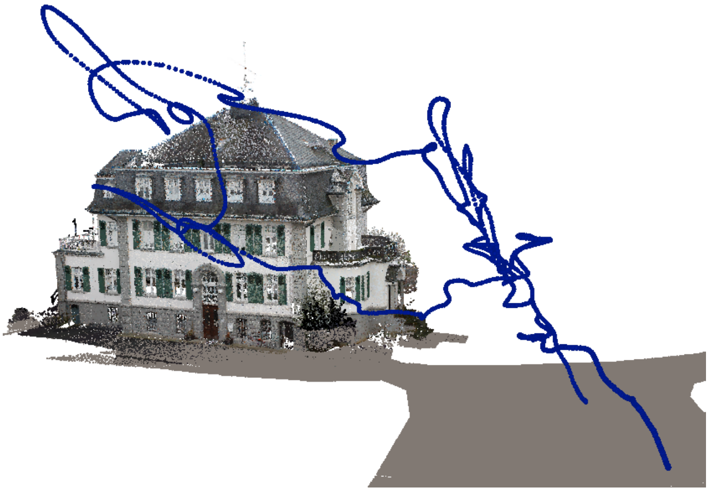


Figure 4.14: Trajectory of the UAV flight overlayed with a georeferenced 3D model of a nearby building.

evaluation. Figure 4.15 shows the theoretical accuracy of the position and orientation of the copter at the estimated keyframes. The rotation angles become more accurate if enough GPS observations constrain the rotation estimation. The highest rotational precision of 0.5° is preserved from the 50th keyframe until landing. The uncertainty is confirmed with the estimated variance factor being in the order of one, assuming an image point accuracy of 2 pixel. As mentioned before, the image point accuracy of 2 pixel may be seen as somewhat low, but results from the frame-wise KLT tracking, see Sec. 4.4.1.

In addition, Figure 4.16 shows the residuals of the GPS DD measurements after optimization, which are within the uncertainties of RTK-GPS solution. As expected, one can also see a larger uncertainty in the height estimate than in the two other directions.

We also evaluate the differences between the 100 Hz Kalman filter solution under GPS-friendly conditions and the bundle adjustment solution. The visual information improves the pose estimate and the differences in each axis direction of the UTM coordinate system between both estimates on average is 1.1 cm in east and north direction and 3.2 cm for the height.

The second experiment is designed to demonstrate the potential of our approach to handle underconstrained GPS situations, i.e. situations in which less than 4 satellites are available. Standard GPS receivers report a GPS loss and cannot estimate a solution. Through the combination of visual SLAM and DD measurements, we can however compute a solution and exploit individual double differences. As can be seen from the trajectories shown in Figure 4.17, exploiting two DD measurements (3 satellites) improves the trajectory estimates substantially and thus is a valuable information for UAV operating in GPS-unfriendly environments.

The last experiment is designed to show the highly accurate georeferenced mapping that is possible using our system. On the copter, we employ a four megapixel camera triggered with 1 Hz for georeferenced mapping, as already mentioned in Sec. 1.2. For near

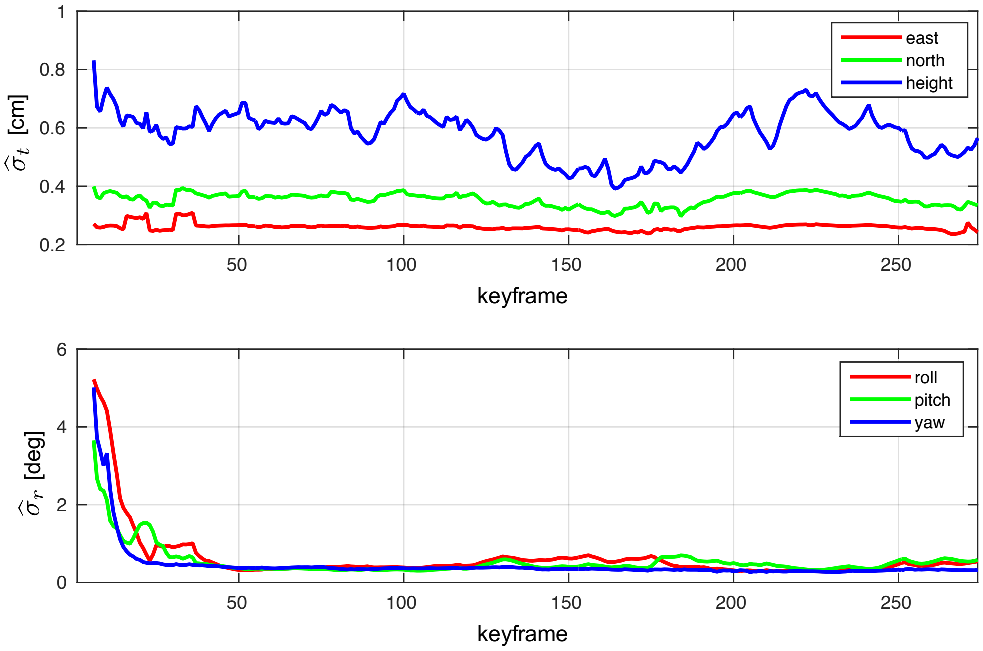


Figure 4.15: Theoretical standard deviation $\widehat{\sigma}_t$ of position and $\widehat{\sigma}_r$ of orientation angles of copter at keyframes. The high long-time precision in the position is provided by the RTK-GPS, the high precision of the rotations is due to the high relative orientation accuracy obtained with bundle adjustment.

real-time georeferenced mapping, the images of the camera are transmitted to a ground station via WiFi together with the onboard computed georeferenced pose of the copter. The ground station runs an incremental bundle adjustment, which integrates the pose computed onboard as prior information.

To obtain the georeferenced poses of the four megapixel camera at the times of image acquisition, the system calibration needs to be known, which we estimate in advance according to the system calibration estimation procedure described in Chap. 3. But instead of using a system self-calibration for the omnidirectional multi-camera system as in the examples of Sec. 3.7.4, we make use of multiple AprilTags with known 3D coordi-

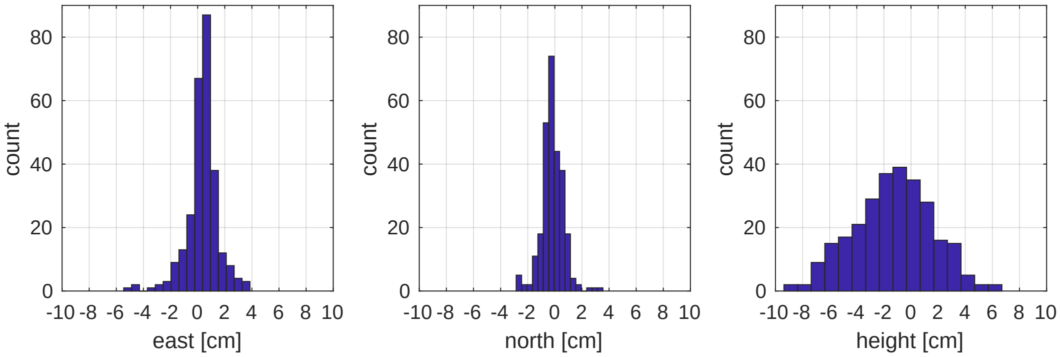


Figure 4.16: Residuals between incrementally estimated positions of keyframes for the GPS double differences.

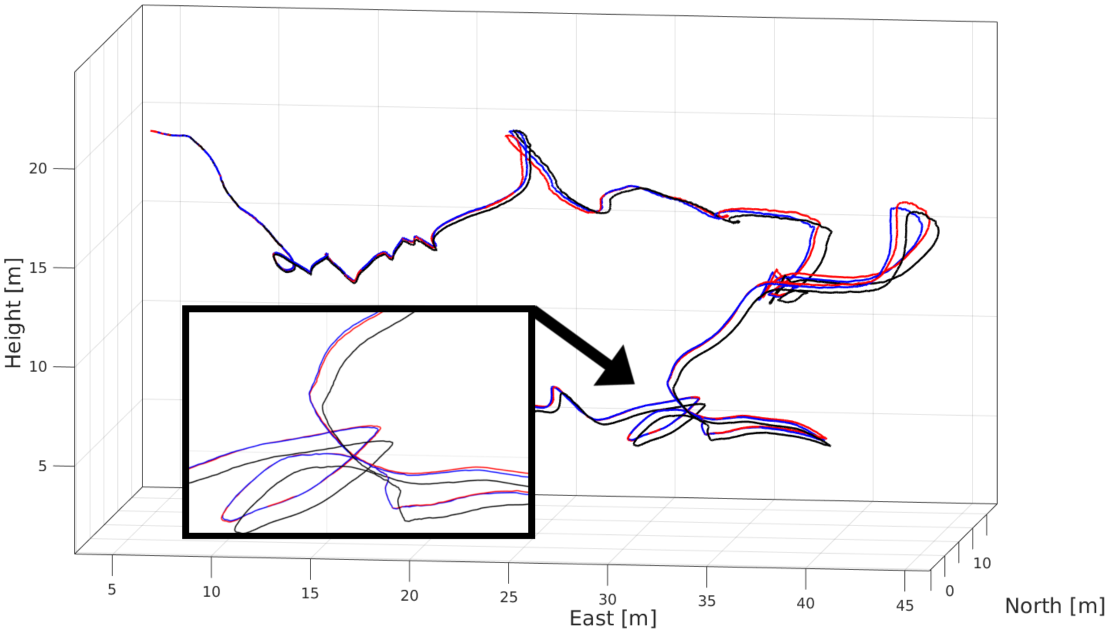


Figure 4.17: The three estimated trajectories show the benefit of operating on raw GPS DD measurements. The red line represents the trajectory exploiting full GPS information (5 - 8 satellites) and is considered as the reference trajectory. Assuming that only 3 satellites are available, the combination of only 2 GPS DD measurements with visual information leads to the solution shown in blue that is much closer to the red reference trajectory than the GPS-free solution shown in black, which could not exploit underconstrained GPS measurements.

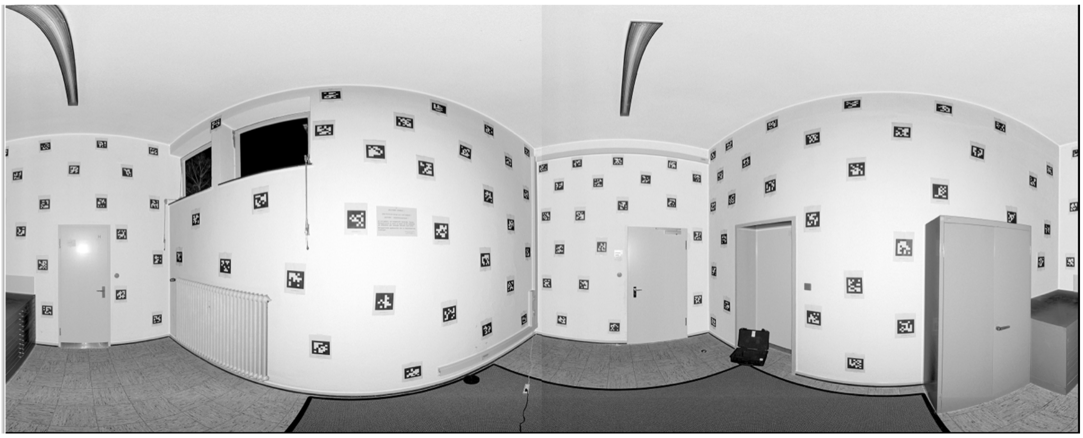


Figure 4.18: The measured intensities of a dense laser scan visualized in a panoramic grayscale image. The image allows to detect each AprilTag attached on the four walls of the room and to obtain the corresponding 3D position from the scanned point cloud.

nates. AprilTags are fiducial markers that can be automatically identified and localized in grayscale images with subpixel precision, see Olson (2011). For the system calibration, we mounted AprilTags on the four walls of an approx. 15 m^2 room and made a highly dense 360° terrestrial laser scan. Figure 4.18 shows the panoramic image which contains the intensity information of each scanned 3D point. To obtain the 3D position of each AprilTag, we detected the AprilTags in the intensity image and extracted the corresponding 3D position from the dense laser scan.

To evaluate the quality of our map estimates, we mapped the house along which the copter flew with a terrestrial laser scanner and precisely georeferenced the point-cloud so that it can be used as a near ground truth 3D model.

As the map built using the high resolution camera on our copter is also georeferenced, both models can be compared without any further alignment. We compare our reconstruction with the georeferenced terrestrial laser scan to evaluate the quality of the determined poses, see Figure 4.19. The median of the absolute differences to the nearest neighbors in each axis direction is around 1 cm. A robust MAD estimation in the component-wise deviations results in about 3 cm and 50 % of all points that have a distance smaller than 5 cm to the nearest neighbor. The full distribution is given in the histogram in Figure 4.19. This experiment shows that our approach generated accurate georeferenced 3D point clouds online.

All computations for the integrated pose estimation are performed onboard the copter, which is equipped with a standard 3.6 GHz Intel CPU with 4 cores and the Kalman filter runs on the real-time board. The tracking for all four cameras and the pose estimation with resection are done at 10 Hz and the keyframe-based optimization runs once per second and is completed before the next keyframe is created and the next optimization would be triggered.

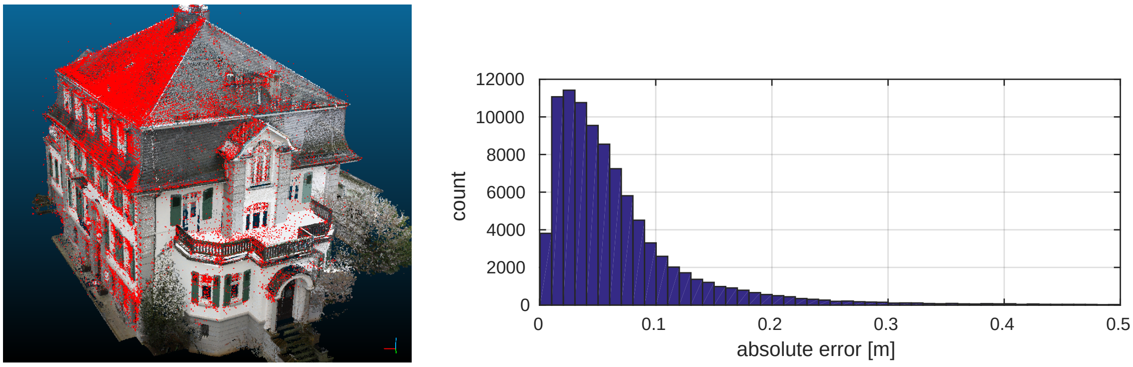


Figure 4.19: Evaluation of reconstructed point cloud with georeferenced terrestrial laser scan. Left: Point cloud from reconstruction using high resolution images (red) and point cloud from terrestrial laser scan (textured). Right: Histogram of the distances between the individual points from the SLAM system and the terrestrial laser scan.

In summary, the experimental evaluation shows that the proposed system offers accurate pose estimation for light-weight UAVs at 10 Hz. Our visual SLAM system can furthermore exploit underconstrained RTK-GPS observations with less than 4 satellites, which reduces the drift in comparison to SLAM systems with traditional GPS integration. Through the effective fusion of GPS, IMU, and visual information, we can compensate GPS-unfriendly situations. Finally, we compared our 3D point cloud to a georeferenced near ground truth 3D model providing an objective measure of the quality of the computed point cloud.

4.5 Conclusion

In this chapter, we presented an effective system for online pose and simultaneous map estimation designed for light-weight UAVs. Our system performs a keyframe-based bundle adjustment in an unknown scene. Incremental bundle adjustment is performed by using the iSAM2 algorithm for sparse nonlinear incremental optimization in combination with our measurement equations that allows for multi-view cameras, omnidirectional cameras and scene points at infinity.

Experiments show the high potential of the incremental bundle adjustment w.r.t. time requirements and optimality. The experiments show that a high accuracy level in the position can be obtained, which is in the order of RTK GPS. Long-time stability and a georeferenced position can be obtained by integrating GPS information. Using fisheye cameras and the inclusion of far points lead to stable poses. The inclusion of GPS is necessary in unknown environments for georeferencing. The visual odometry can bridge gaps due to interruption of the GPS signal with high accuracy.

In addition to that, we presented an effective bundle adjustment solution exploiting RTK-GPS carrier phase observations, IMU data and visual SLAM in an incremental fashion at 10 Hz. The overall system yields a robust pose estimate at high frequencies and can handle underconstrained GPS situations effectively. The components have been implemented as ROS modules and the software runs online on our 5 kg multi-copter. By comparing our results with models generated from georeferenced terrestrial laser scanners, we show a deviation of the median to our point clouds of less than 1 cm.

5 Quality of Dense Stereo with Fisheye Cameras

So far, we have focused on recovering the pose and a sparse scene reconstruction on the basis of an image sequence of a camera system. In this chapter, we investigate the potential of dense stereo reconstruction using a stereo camera with fisheye lenses.

The contribution of this chapter is an approach for re-using existing dense stereo methods with fisheye cameras. For this, we follow the approach of Abraham and Förstner (2005) and generate virtual stereo image pairs that can then be used with existing dense stereo methods that assume the epipolar lines to correspond to a row in the image. This has the great advantage that highly optimized existing dense stereo methods can be applied as a black box without modifications, even with cameras that have a field of view of more than 180°. In this thesis, we consider semi-global matching (SGM) by Hirschmüller (2008) and efficient large-scale stereo (ELAS) by Geiger et al. (2010) but our approach is not restricted to these dense stereo methods. Our approach transfers to other stereo methods which provide a dense disparity image, as for example the more recent CNN-based matching approaches by Li et al. (2018), Li et al. (2017) or Taniai et al. (2017) which incorporate the CNN-based matching cost function of Žbontar and LeCun (2016).

Using the obtained disparity image, we derive a dense 3D point cloud together with the uncertainty of each single 3D point. We provide a detailed accuracy analysis of the obtained dense stereo results. This requires a realistic stochastic model for the disparities of the matched image points. The core of this chapter therefore is a rigorous variance component estimation that optimally estimates the variance of the disparity at a point as a function of the distance of that image point to the image center and thus allows to predict the accuracy of the 3D points. We evaluate the significance of the improved stochastic model on scene reconstruction.

5.1 Introduction

Using pairs of fisheye cameras allows to capture a large field of view stereoscopically. In contrast to classical cameras, however, fisheye lenses do not follow a perspective projection and cannot be approximated well using the pinhole camera model. This holds especially for cameras with a field of view of more than 180° and this often prevents the use of methods that assume a perspective projection model.

This chapter deals with computing dense stereo information from fisheye cameras and

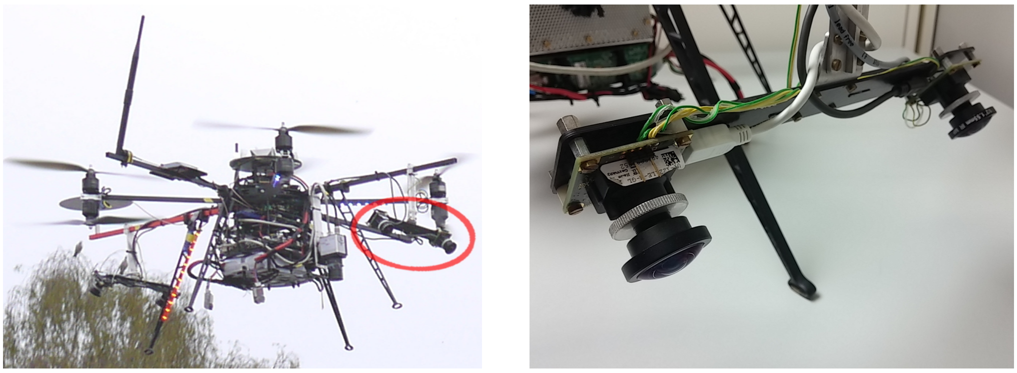


Figure 5.1: Our UAV (left) equipped with fisheye stereo cameras with an opening angle of 185°. This chapter describes how dense fisheye stereo can be computed based on existing methods for perspective cameras and analyzes the accuracy of the obtained point cloud from a theoretical and experimental perspective. The overall system runs at 6 - 7 Hz on our copter and provides 3D point clouds including information about its accuracy to improve reconstruction.

provides a detailed analysis of the quality of the recovered 3D points with respect to the fisheye specific light projection on the image planes.

Traditional approaches to stereo vision rely on sparse points for which the 3D position is estimated through triangulation. The availability of sparse depth data only leads to more difficult object segmentation (van der Mark and Gavrila, 2006), scene understanding, or obstacle detection tasks. Thus, there is an increasing interest in semi-dense and dense reconstruction approaches (Engel et al., 2013) with applications in transportation systems (van der Mark and Gavrila, 2006), autonomous cars (Franke et al., 2013), or unmanned aerial vehicles (Schmid et al., 2014).

A central task in sparse as well as dense stereo methods is to identify correspondences between the image pairs. By exploiting the epipolar geometry, we can reduce the 2D search problem to a simpler 1D problem. Depending on the used projection model for calibration and rectification, this 1D space corresponds to a straight line in a perspective projection or to a more complicated curve, e.g. a circular line in a stereographic projection (Heller and Pajdla, 2009). Most systems for dense stereo assume that this 1D space is a straight line in the image, sometimes even that this line corresponds to a row in the image. This assumption can prevent the direct use of wide-angle or fisheye cameras with black-box dense stereo algorithms.

5.2 Related Work

Stereo matching is a large research area and a substantial number of algorithms for identifying stereo correspondence has been proposed. A good overview is given by Scharstein and Szeliski (2002). Over the last decade, more dense stereo and reconstruction methods have been developed. Popular approaches include semi-global matching by Hirschmüller

(2008) and efficient large-scale stereo by Geiger et al. (2010). Zbontar and LeCun (2015) apply a convolutional neural network to the problem of stereo matching and achieve accurate results on several benchmark datasets.

Most of the dense stereo techniques have been designed for perspective cameras and cannot directly deal with the input of fisheye cameras. Thus, Gao and Shen (2017) rectify an image pair of a fisheye stereo camera into four image pairs following the perspective projection model to apply such dense stereo methods. The idea of combining fisheye camera calibration and epipolar rectification for stereo computations goes back to Abraham and Förstner (2005), who presented a method that can be seen as a specialization of the work by Pollefeys et al. (1999). Esparza et al. (2014) use a modified version of the epipolar rectification model to allow for wide stereo bases and largely disaligned optical axes. They apply epipolar rectification only on the overlapping image parts, which allows fast matching of detected keypoints along the image rows. Other rectification approaches exist, for example for binocular cylindrical panoramic images (Ishiguro et al., 1992), which limit the vertical field of view and do not lead to epipolar images.

A review of fisheye projection models is given by Abraham and Förstner (2005). The work also provides an approach to calibrate fisheye stereo camera systems. Tommaselli et al. (2014) showed that all the projection models in (Abraham and Förstner, 2005) are equally suitable to model fisheye cameras by comparing the residuals in 3D reconstruction after calibration. Fu et al. (2014) determine the intrinsic and extrinsic parameters of a camera system that can consist of many overlapping fisheye cameras by using a wand with three collinear feature points and provide a toolbox online. Calibration approaches for a camera system with non-overlapping fisheye cameras are described in (Schneider and Förstner, 2013) and (Heng et al., 2014), both approaches use bundle adjustment without the need of fiducial markers.

Wang et al. (2015) give a formula to calculate the loss of spatial resolution of a fisheye camera with increasing distance to the image center. Their approach improves the image quality in regions with small spatial resolution using compressive sensing assuming an equi-distance projection model (Xu and Peng, 2014), but they do not provide a rigorous statistical analysis of their results.

Computing stereo information from fisheye cameras has also been investigated by other researchers. For example, Kita (2011a) analyzes dense 3D measurements obtained with a fisheye stereo camera pair with perfect calibration, observing the workspace of a humanoid robot. Herrera et al. (2011) propose a strategy for obtaining a disparity map from hemispherical stereo images captured with fisheye lenses in forest environments. To support the dense stereo process, they segment and classify the textures in the scene and consider only those matches belonging to the same class. Also Moreau et al. (2013) address dense 3D point cloud computation with fisheye stereo pairs using epipolar curves with a unit sphere model. Arfaoui and Thibault (2015) use cubic spline functions to model tangential and radial distortions in panoramic stereo vision systems to simplify stereo matching.

They also provide the mathematical relationship between matches to determine 3D point locations. Häne (2016) extends the plane-sweeping stereo matching of Gallup et al. (2007) for fisheye cameras by incorporating the fisheye projection model directly into the plane-sweeping stereo matching algorithm, which eliminates the need of prior rectification, but needs to employ a GPU to achieve real-time capability.

Compared to our approach, neither Kita, Herrera et al., Moreau et al., Arfaoui et al. nor Gallup et al. can exploit existing dense stereo implementations as a black box. Furthermore, they do not provide a detailed analysis of the accuracy of their results.

In addition to the dense stereo approaches, several new dense 3D reconstruction systems have been proposed in recent years, for example Dense Tracking and Mapping by Newcombe et al. (2011) or the approach by Stühmer et al. (2010) that computes a dense reconstruction using variational methods. The simultaneous optimization of dense geometry and camera parameters is possible but is a rather computationally intensive task (Aubry et al., 2011). In order to deal with the computational complexity for real-time operation, semi-dense approaches are becoming increasingly popular, e.g.by Engel et al. (2013) even for monocular cameras.

Visual 3D reconstruction received also quite some attention in the context of light-weight UAV systems over the past few years. Especially in this application, light-weight sensors with a large field of view are attractive due to the strong payload limitations. For example Pizzoli et al. (2014) propose a dense reconstruction approach for UAVs. They work with a single perspective camera and their approach combines Bayesian estimation and convex optimization performing the reconstruction on a GPU at frame rate. Related to that, combinations of perspective monocular cameras on an indoor UAV and RGB-D cameras on a ground vehicle have been used for simultaneous localization and mapping tasks aligning the camera information with dense ground models (Forster et al., 2013). In contrast, our method allows for using dense stereo methods with fisheye cameras on UAVs and provides an estimate of the accuracy of the returned point-cloud.

5.3 Two Popular Dense Stereo Methods for Perspective Cameras

In our work, we consider two popular dense stereo methods for computing a dense depth reconstruction given a stereo pair. These two methods are efficient large-scale stereo (ELAS) by Geiger et al. (2010) and semi-global matching (SGM) by Hirschmüller (2008). Both have been designed for calibrated perspective cameras and the output of both methods is a disparity image.

ELAS computes disparity maps from rectified stereo image pairs and are robust against moderate illumination changes. ELAS provides a generative probabilistic model for stereo matching, which allows for dense matching using small aggregation windows. The Bayesian approach builds a prior over the disparity space by forming a triangulation on a set of robustly matched correspondences, so-called support points. ELAS applies a maximum a-posteriori estimation scheme to compute the disparities given all observations in the other image which are located on the given epipolar line. This yields an efficient algorithm with near real-time performance that also allows for parallelization.

Semi-Global matching aims at combining local and global techniques in order to obtain an accurate, pixel-wise matching at comparably low computational requirements. It uses mutual information as the matching cost for corresponding points and the global radiometric difference are modeled in a joint histogram of corresponding intensities. An extension of SGM relies on the Census matching cost. Census is slightly inferior to mutual information if there are only global radiometric differences but it has been shown to outperform mutual information in the presence of local radiometric changes and thus is beneficial in most real-world applications (Hirschmüller, 2011).

SGM uses a global cost function that penalizes small disparity steps, which are often part of slanted surfaces, less than real discontinuities. The cost function is optimized similarly to scan-line optimization and it finds an efficient solution for the 1D case. The key idea in SGM is to perform this computation along eight straight line paths ending in the pixel considering symmetry from all directions. Each path encodes cost for reaching the pixel with a certain disparity. For each pixel and each disparity, the costs are summed over the eight paths and at each pixel, the disparity with the lowest cost is chosen.

5.4 Dense Fisheye Stereo and its Accuracy

This section describes our approach to obtain a dense 3D point cloud together with its uncertainty information using a stereo camera with fisheye lenses. In its first two subsections, it describes the fisheye-specific light projection and the epipolar rectification model for fisheye cameras proposed in Abraham and Förstner (2005) that makes common dense stereo methods applicable. The third subsection describes how we compute the dense 3D point cloud with its uncertainty through variance propagation using the disparity information.

5.4.1 Fisheye Model

The fisheye specific projection from a 3D ray to a 2D image point can be described using the equi-distance projection model, a reasonable first-order approximation for the intrinsically non-perspective projection of fisheye lenses (Xiong and Turkowski, 1997). The equi-distance projection model projects a 3D camera ray ${}^c\boldsymbol{x} = [{}^cx, {}^cy, {}^cz]^\mathsf{T}$ in the camera reference frame (indicated by superscript c), whose orientation is specified by the two angles ϕ and α as depicted in Figure 2.3 on page 28, into a 2D position

$${}^i\boldsymbol{x} = \begin{bmatrix} {}^ix \\ {}^iy \end{bmatrix} = \begin{bmatrix} \frac{\text{atan2}({}^cr,{}^cz)}{{}^cr}\,{}^cx \\ \frac{\text{atan2}({}^cr,{}^cz)}{{}^cr}\,{}^cy \end{bmatrix} = \sin\phi \begin{bmatrix} \cos\alpha \\ \sin\alpha \end{bmatrix} \tag{5.1}$$

with ${}^{c}r = \sqrt{{}^{c}x^2 + {}^{c}y^2}$. The equi-distance projection basically differs from the perspective projection in the definition of the radial projection function ${}^{i}r(\phi)$. The radial distance in the conditioned image ${}^{i}r = \sqrt{{}^{i}x^2 + {}^{i}y^2} = \phi$ only depends on the angle ϕ between the 3D ray ${}^{c}\boldsymbol{x}$ and the optical axis and becomes a monotonously increasing function, which allows for a field of view larger than 180°.

The relation of conditioned image point ${}^{i}\boldsymbol{x}$ to its unconditioned coordinates is given by $\boldsymbol{x}' = c\,{}^{i}\boldsymbol{x} - \boldsymbol{h}$ with the principal point $\boldsymbol{h} = [h_x, h_y]^{\mathsf{T}}$ and the principal distance c obtained by camera calibration, e.g. according to Abraham and Förstner (2005). Given a 2D point ${}^{i}\boldsymbol{x}$, the inverse transformation of Eq. (5.1) into a 3D camera ray reads as

$$ {}^{c}\boldsymbol{x} = [{}^{c}x, {}^{c}y, {}^{c}z]^{\mathsf{T}} = \left[\frac{\sin {}^{i}r}{{}^{i}r}{}^{i}x,\ \frac{\sin {}^{i}r}{{}^{i}r}{}^{i}y,\ \cos {}^{i}r\right]^{\mathsf{T}}. \tag{5.2} $$

In Sec. 5.4.3, we will use this model to propagate the positional uncertainty of an observed image point to its corresponding camera ray. Note that we have not introduced additional parameters for lens distortion and assume them to be negligibly small after proper calibration.

5.4.2 Epipolar Rectification

In a camera pair with two projection centers, all epipolar planes intersect at the baseline. Despite ideal properties of the stereo cameras, like parallel optical axis, the introduced equi-distance projection model does not lead to images where each 3D point is projected into the same row in both cameras, thus the epipolar lines are curved. To obtain parallel epipolar lines such that the vertical disparity vanishes and the correspondence search can be reduced to a one-dimensional search along the image rows, we use the epipolar rectification model proposed by Abraham and Förstner (2005). The epipolar rectification model is a special case of the general rectification model given in Pollefeys et al. (1999). Other projection models exist but limit the vertical field of view as for example the binocular cylindrical projection model proposed by Ishiguro et al. (1992) or lead to multiple projection centers for each camera as omnivergent stereo proposed by Seitz et al. (2002). Applying the epipolar projection model requires a calibrated stereo camera with intrinsic and relative calibration. We exploit the concept of a virtual camera to achieve a rectification for the image pair that is independent of the real projection system and leads to ideal properties: identical interior orientation with no distortions, no camera rotation and a baseline in one axis direction. The epipolar equi-distance rectification model projects the epipolar planes to the same image row in both images.

The projection function is given by

$$ {}^{i}\boldsymbol{x} = \begin{bmatrix} \operatorname{atan2}\left({}^{c}x, \sqrt{{}^{c}y^2 + {}^{c}z^2}\right) \\ \operatorname{atan2}\left({}^{c}y, {}^{c}z\right) \end{bmatrix} = \begin{bmatrix} \beta \\ \psi \end{bmatrix} \tag{5.3} $$

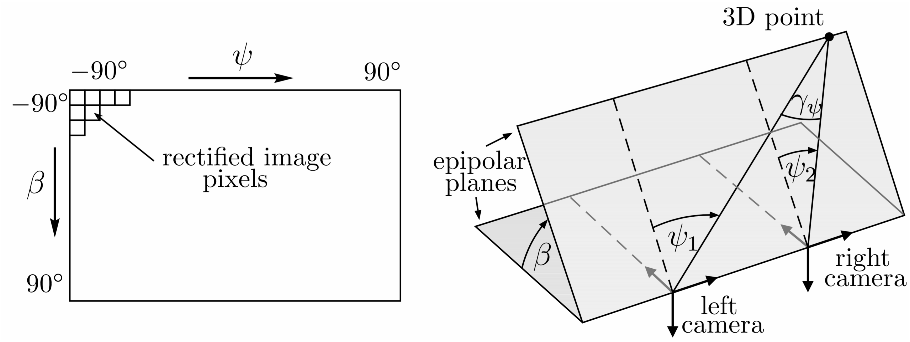


Figure 5.2: The projection of the epipolar planes inside the rows according to Eq. (5.3). Each pixel coordinate of rectified image corresponds directly to the angles β and ψ. Figure adapted from Abraham and Förstner (2005).

where the coordinates of the conditioned image point ${}^{i}\boldsymbol{x}$ correspond directly to the angles ψ and β that describe the ray to the observed 3D point as shown in Figure 5.2: β characterizes the pitch angle of each epipolar plane and ψ characterizes the projection inside the epipolar plane, i.e. the image row.

For image rectification, principal distance c and principal point $\boldsymbol{h}$ from calibration can be used. Given an image pixel position $\boldsymbol{x}'$ in the rectified image the corresponding angles are then obtained by the relation $[\beta, \psi]^{\mathsf{T}} = (\boldsymbol{x}' - \boldsymbol{h})/c$.

The transformation from conditioned image position ${}^{i}\boldsymbol{x}$ into a ray direction ${}^{c}\boldsymbol{x}$ with unit length is given by

$$
{}^{c}\boldsymbol{x} = \begin{bmatrix} \sin {}^{i}x \\ \cos {}^{i}x \sin {}^{i}y \\ \cos {}^{i}x \cos {}^{i}y \end{bmatrix} . \tag{5.4}
$$

5.4.3 3D Point Cloud with Uncertainty

We derive the 3D point coordinates with their uncertainty through variance propagation given an image point with its disparity information. Let $\Sigma_{x'x'}$ describe the positional uncertainty of the image point $\boldsymbol{x}' = [x', y']^{\mathsf{T}}$ in the *unrectified image.*

For the fisheye lenses, we use the equi-distant camera model according to Sec. 5.4.1. Using the principal distance c and principal point $\boldsymbol{h}$ from calibration, we obtain the conditioned image coordinates ${}^{i}\boldsymbol{x}$ with their covariance matrix $\Sigma_{{}^{i}x{}^{i}x}$ as

$$
{}^{i}\boldsymbol{x} = \frac{1}{c}\left(\boldsymbol{x}' - \boldsymbol{h}\right) \qquad \text{and} \qquad \Sigma_{{}^{i}x{}^{i}x} = \frac{1}{c^2}\,\Sigma_{x'x'} . \tag{5.5}
$$

This yields the corresponding camera ray ${}^{c}\boldsymbol{x}$ according to Eq. (5.2) and its covariance matrix through variance propagation

$$
\Sigma_{{}^{c}x{}^{c}x} = J_1 \Sigma_{{}^{i}x{}^{i}x} J_1^{\mathsf{T}} \tag{5.6}
$$

with

$$\mathsf{J}_1 = \begin{bmatrix} \dfrac{\sin({}^ir){}^iy^2 + \cos({}^ir){}^ix^2\,{}^ir}{({}^ix^2 + {}^iy^2)^{3/2}} & \dfrac{(\cos({}^ir){}^ir - \sin({}^ir)){}^iy{}^ix}{({}^ix^2 + {}^iy^2)^{3/2}} \\ \dfrac{(\cos({}^ir){}^ir - \sin({}^ir)){}^iy{}^ix}{({}^ix^2 + {}^iy^2)^{3/2}} & \dfrac{\cos({}^ir){}^iy^2\,{}^ir + \sin({}^ir){}^ix^2}{({}^ix^2 + {}^iy^2)^{3/2}} \\ -\dfrac{\sin({}^ir){}^ix}{{}^ir} & -\dfrac{\sin({}^ir){}^iy}{{}^ir} \end{bmatrix}. \tag{5.7}$$

Given the previously defined rectification, we obtain the angles ψ and β from a ray ${}^c\boldsymbol{x}$ according to Eq. (5.3) and for the covariance matrix follows

$$\Sigma_{\left[\begin{smallmatrix}\beta\\ \psi\end{smallmatrix}\right]} = \mathsf{J}_2 \Sigma_{{}^cx{}^cx} \mathsf{J}_2^{\mathsf{T}} \tag{5.8}$$

with

$$\mathsf{J}_2^{\mathsf{T}} = \begin{bmatrix} \dfrac{\sqrt{{}^cy^2 + {}^cz^2}}{{}^cx^2 + {}^cy^2 + {}^cz^2} & 0 \\ \dfrac{-{}^cx{}^cy}{\sqrt{{}^cy^2 + {}^cz^2}({}^cx^2 + {}^cy^2 + {}^cz^2)} & \dfrac{{}^cz}{{}^cy^2 + {}^cz^2} \\ \dfrac{-{}^cx{}^cz}{\sqrt{{}^cy^2 + {}^cz^2}({}^cx^2 + {}^cy^2 + {}^cz^2)} & \dfrac{-{}^cy}{{}^cy^2 + {}^cz^2} \end{bmatrix}. \tag{5.9}$$

As the corresponding camera rays do intersect in one point (as β is identical for both rays), we can determine its coordinates easily. Let s be the distance from the left camera along the camera ray ${}^c\boldsymbol{x}$ to the unknown 3D point $\boldsymbol{p} = s\,{}^c\boldsymbol{x}$. Camera ray ${}^c\boldsymbol{x}$ can be derived with β and ψ according to Eq. (5.4). To compute s, we use the angles β and ψ and the ψ-disparity γ_ψ given with the image coordinates of corresponding points, see also Figure 5.2. Note that the apical angle, i.e. the intersection angle, complies with the disparity angle

$$\gamma_\psi = \gamma_{x'}/c \tag{5.10}$$

with the measured disparity $\gamma_{x'}$ in the epipolar rectified image and the principal distance c used for this rectification. This can be shown using the angular sum $\gamma_\psi = 180° - \psi'_1 - \psi'_2$ with the interior angles $\psi'_1 = 90° - \psi$ and $\psi'_2 = 90° + \psi - \gamma_\psi$.

Exploiting the law of sines, we obtain

$$s = b\,\frac{\sin(90° + \psi - \gamma_\psi)}{\sin\gamma_\psi} = b\,\frac{\cos(\psi - \gamma_\psi)}{\sin\gamma_\psi}, \tag{5.11}$$

with b being the base line, which leads to the 3D coordinates of the point $\boldsymbol{p}$ as

$$\boldsymbol{p}(\psi, \beta, \gamma_\psi) = b \, \frac{\cos(\psi - \gamma_\psi)}{\sin \gamma_\psi} \begin{bmatrix} \sin \psi \\ \cos \psi \sin \beta \\ \cos \psi \cos \beta \end{bmatrix} . \tag{5.12}$$

The covariance matrix of $\boldsymbol{p}$ is obtained through

$$\Sigma_{pp} = \mathsf{J}_3 \, \mathrm{Diag}([\Sigma_{\left[\begin{smallmatrix}\beta\\ \psi\end{smallmatrix}\right]}, \sigma^2_{\gamma_\psi}]) \, \mathsf{J}_3^{\mathsf{T}} \quad \text{with} \quad \mathsf{J}_3 = \left[\frac{\partial \boldsymbol{p}}{\partial \psi}, \frac{\partial \boldsymbol{p}}{\partial \beta}, \frac{\partial \boldsymbol{p}}{\partial \gamma_\psi}\right] . \tag{5.13}$$

The three column vectors of J_3 are the partial derivatives of Eq. (5.12) and read as

$$\frac{\partial \boldsymbol{p}}{\partial \psi} = b \begin{bmatrix} \dfrac{\cos(\psi - \gamma_\psi)\cos(\psi) - \sin(\psi - \gamma_\psi)\sin(\psi)}{\sin(\gamma_\psi)} \\ 0 \\ \dfrac{\sin(\psi)\,(\cos(\psi - \gamma_\psi)\cos(\gamma_\psi) - \sin(\psi - \gamma_\psi)\sin(\gamma_\psi))}{1 - \cos^2(\gamma_\psi)} \end{bmatrix} , \tag{5.14}$$

$$\frac{\partial \boldsymbol{p}}{\partial \beta} = b \begin{bmatrix} \dfrac{-\sin(\beta)\,(\sin(\psi - \gamma_\psi)\cos(\psi) + \cos(\psi - \gamma_\psi)\sin(\psi))}{\sin(\gamma_\psi)} \\ \dfrac{\cos(\psi - \gamma_\psi)\cos(\psi)\cos(\beta)}{\sin(\gamma_\psi)} \\ \dfrac{\cos(\psi)\sin(\beta)\,(\sin(\psi - \gamma_\psi)\sin(\gamma_\psi) - \cos(\psi - \gamma_\psi)\cos(\gamma_\psi))}{1 - \cos^2(\gamma_\psi)} \end{bmatrix} , \tag{5.15}$$

$$\frac{\partial \boldsymbol{p}}{\partial \gamma_\psi} = b \begin{bmatrix} \dfrac{-\cos(\beta)\,(\sin(\psi - \gamma_\psi)\cos(\psi) + \cos(\psi - \gamma_\psi)\sin(\psi))}{\sin(\gamma_\psi)} \\ \dfrac{-\cos(\psi - \gamma_\psi)\cos(\psi)\sin(\beta)}{\sin(\gamma_\psi)} \\ \dfrac{\cos(\psi)\cos(\beta)\,(\sin(\psi - \gamma_\psi)\sin(\gamma_\psi) - \cos(\psi - \gamma_\psi)\cos(\gamma_\psi))}{1 - \cos^2(\gamma_\psi)} \end{bmatrix} . \tag{5.16}$$

5.5 Improved Stochastic Observation Model

We start with a *standard stochastic model* for the observed entities. The sensor coordinates of the images points are assumed to be identically and independently distributed $\mathbb{D}([x'_i, y'_i]^{\mathsf{T}}) = \sigma^2_x \, \mathsf{I}_2$ and the disparities are assumed to have the same variance $\mathbb{D}(\gamma_\psi) = \sigma^2_{\gamma_\psi}$. Due to the properties of the optics, we can expect that in a first approximation the accuracy of the sensor coordinates depends on the angle ϕ between the viewing direction and the direction to the scene point.

In order to determine this dependency, we observe planar surfaces in a scene and analyze the residuals using a robust version of variance component analysis leading to a refined or *improved stochastic model* for the observation's variances. Using a stochastic model which is closer to reality should lead to better estimates of the plane's parameters. We will check this empirically by analyzing orthogonal planes.

5.5.1 Variance Analysis

Classical estimation procedures assume the covariance matrix Σ_{ll} of the $n = 1, ..., N$ observations to be known up to an unknown variance factor, where $\boldsymbol{l}$ refers to the observations. Thus, the stochastic model is assumed to be $\Sigma_{ll} = \sigma_0^2 \Sigma_{ll}^a$, where Σ_{ll}^a is an approximation for the covariance matrix, and the unknown variance factor σ_0^2 is assumed to be one. Based on a Gauss–Markov model of the form

$$p(\boldsymbol{l}) = \mathcal{N}(A\boldsymbol{x} + \boldsymbol{a}, \sigma_0^2 \Sigma_{ll}^a) \tag{5.17}$$

with the Jacobian A and U unknown parameters, we obtain the ML-estimate

$$\widehat{\boldsymbol{x}} = \Sigma_{\widehat{x}\widehat{x}} A^\mathsf{T} \Sigma_{ll}^{-1} (\boldsymbol{l} - \boldsymbol{a}) \tag{5.18}$$

with the covariance matrix

$$\Sigma_{\widehat{x}\widehat{x}} = (A^\mathsf{T} \Sigma_{ll}^{-1} A)^{-1}. \tag{5.19}$$

With the estimated residuals $\widehat{\boldsymbol{v}} = A\widehat{\boldsymbol{x}} + \boldsymbol{a} - \boldsymbol{l}$ and the redundancy $R = N - U$, we have the unbiased estimated variance factor

$$\widehat{\sigma}_0^2 = \widehat{\boldsymbol{v}}^\mathsf{T} \Sigma_{ll}^{-1} \widehat{\boldsymbol{v}} / R \qquad \text{with} \qquad \sigma_{\widehat{\sigma}_0} = \sqrt{2/R}\, \sigma_0 \,. \tag{5.20}$$

For an *improved stochastic model*, we now assume that the variances of the observations follow the model

$$\Sigma_{ll} = \sum_{j=1}^{J} \sigma_j^2 \Sigma_j^a \tag{5.21}$$

with known approximate covariance matrices and unknown variance factors, also called variance components, σ_j^2. In our case, we assume

$$\sigma_{l'_n}^2 = \sigma_1^2 + \sigma_2^2 \phi_n^{2p} \,, \tag{5.22}$$

i.e. the noise of the sensor coordinates is a sum of a constant noise term $\underline{n}_1$ with $p(n_1) = \mathcal{N}(0, \sigma_{01}^2)$ and a noise term $\underline{n}_2$ proportional to the p-th power ϕ_n^p of the angle ϕ_n referring to the n-th observation, thus $p(n_2) = \mathcal{N}(0, \sigma_{02}^2)$. As we will illustrate in the experimental evaluation through the analysis of the variance factors computed for different angles ϕ, this models describes the noise in relation to ϕ well.

This leads to the two covariance matrices

$$\Sigma_1^a = I_N \qquad \text{and} \qquad \Sigma_2^a = \mathrm{Diag}([\phi_n^{2p}])\,. \tag{5.23}$$

With the weight or precision matrix $W_{ll} = \Sigma_{ll}^{-1}$ of the observations and the covariance matrix $\Sigma_{\widehat{v}\widehat{v}} = \Sigma_{ll} - A^{\mathsf{T}}\Sigma_{\widehat{x}\widehat{x}}A$, the general and the specific expressions for the estimated variance components are

$$\widehat{\sigma}_j^2 = \frac{\widehat{\boldsymbol{v}}^{\mathsf{T}} W_{ll}\Sigma_j^a W_{ll}\widehat{\boldsymbol{v}}}{\mathrm{tr}\,(W_{ll}\Sigma_j^a W_{ll}\Sigma_{\widehat{v}\widehat{v}})}\,. \tag{5.24}$$

In our case, this simplifies to the relations

$$\widehat{\sigma}_1^2 = \frac{\sum_n w_n^2 \widehat{v}_n^2}{\sum_n w_n^2 \sigma_{\widehat{v}_n}^2} \qquad \text{and} \qquad \widehat{\sigma}_2^2 = \frac{\sum_n w_n^2 \widehat{v}_n^2 \phi_n^{2p}}{\sum_n w_n^2 \sigma_{\widehat{v}_n}^2 \phi_n^{2p}}\,. \tag{5.25}$$

The estimated variance factors lead to an updated covariance matrix of the observations as in Eq. (5.21) and we apply the estimation procedure iteratively until convergence.

Unfortunately, these relations are not robust and therefore we proceed differently. We use the standardized residuals

$$z_n = \frac{\widehat{v}_n}{\sigma_{\widehat{v}_n}} \sim \mathcal{N}(0,1)\,. \tag{5.26}$$

In our case with $N \ll U$, these values can be safely approximated by $\widehat{\boldsymbol{v}}/\sigma_{l_n}$. Their variance should be close to one, if the model holds. We therefore robustly determine the variance of the residuals for narrow ranges of ϕ, by partitioning the set of all ϕ_n in K bins and use $\widehat{\sigma_z} = 1.48\,\mathrm{MAD}$ with the median absolute difference (MAD) of the z_n in each bin, see Koch (1999) for details. We choose $K = 30$, such that for typical $N < 10000$, the number of values for estimating the variances is larger than 300. From the pairs $\{\mu_{\phi_k}, \widehat{\sigma}_{z,k}^2\}$, where μ_{ϕ_k} is the center of the k-th bin, we determine the variance components σ_j^2 in Eq. (5.25) by simple regression, which is justified as the bins contain the same number N/K of observations.

5.5.2 Orthogonality Improvement

The improved stochastic model should lead to better estimates of the plane parameters. In case of mutually orthogonal planes, the angle ω between the estimated normal directions should get closer to 90° than when using the classical stochastic model.

Estimating the orthogonal planes N times using different stereo images leads to $n = 1, ..., N$ deviations $\omega_n - 90°$. The empirical variance $\widehat{\sigma}_\omega^2 = \frac{1}{N}\sum_n(\omega_n - 90°)^2$ and the theoretical variance σ_ω^2 derived from covariance matrix $\Sigma_{\widehat{\boldsymbol{x}}\widehat{\boldsymbol{x}}}$ of both estimated planes should (a) indicate a higher precision than when using the classical model and (b) confirm empirically the plausibility of the stochastic model if $\widehat{\sigma}_\omega$ and σ_ω comply with the relative

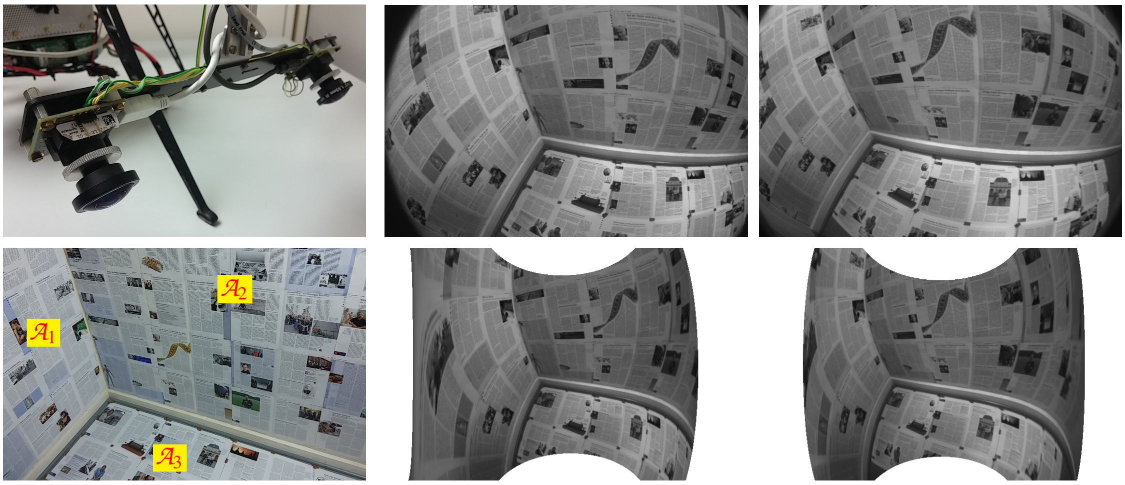


Figure 5.3: Left images: Stereo camera with fisheye lenses and highly textured and mutually orthogonal planes $\mathcal{A}_1$, $\mathcal{A}_2$ and $\mathcal{A}_3$ used for variance analysis. Upper right images: Stereo image pair. Lower right: Image pair after epipolar rectification. Note that all epipolar lines of the left and right image are in the same row.

accuracy of Eq. (5.20), i.e. if $\widehat{\sigma}_\omega/\sigma_\omega \approx \sqrt{2/N}$ holds.

5.6 Experimental Evaluation

The goal of this experimental evaluation is to illustrate that dense fisheye stereo can be achieved and to investigate the accuracy of dense stereo with fisheye cameras using the epipolar rectification model. For the evaluation, we use a stereo camera with a basis of 20 cm and Lensagon BF2M14420 fisheye lenses with a field of view of 185°. We calibrate the stereo camera by estimating the interior and relative orientation according to Abraham and Förstner (2005) using the epipolar equi-distance rectification model. For epipolar rectification, we use a camera constant of $c = 200$ pixel to keep most of the image content in 752×480 images. After rectification the disparity between corresponding points is limited to the same image row, see Figure 5.3.

5.6.1 Variance Analysis

For the first two sets of experiments, we use three highly textured and mutually orthogonal planar surfaces, see Figure 5.3, for evaluating the variance analysis described in Sec. 5.5. To analyze the accuracy of the observations in dependency of the angle ϕ, we capture the three planar surfaces under 30 different poses such that the planes are visible over a broad spectrum of ϕ. For each image pair, we use ELAS and SGM to determine dense disparity information. We use the default settings for robotic environments for ELAS and the default settings for SGM.

For each pixel with disparity information, we obtain the coordinates of a 3D point $\boldsymbol{p}$ in camera frame using Eq. (5.12). We compute the covariance matrix Σ_{pp} according

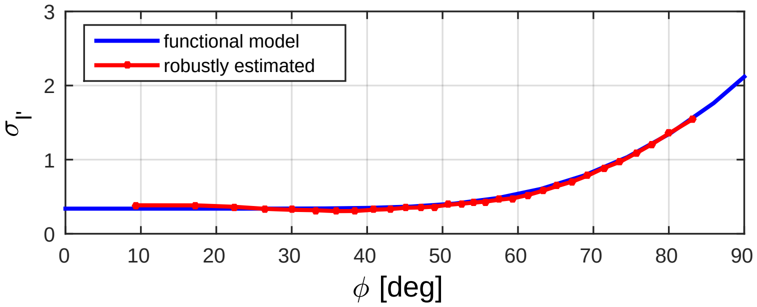


Figure 5.4: The red dots show 30 robust estimates for standard deviation $\widehat{\sigma}_{l'}$ using the residuals of narrow ranges of ϕ, the blue line shows the estimated functional model of $\widehat{\sigma}_{l'}$ over angle ϕ.

to Eq. (5.13) using a standard stochastic model with identically and independently distributed image points and disparities $\sigma_{\gamma_{x'}} = \sigma_{x'} = \sigma_{y'} = 1\,$pixel. We then estimate for each of the 30 captured stereo pairs the three normal directions of the three planes $\mathscr{A}_1$, $\mathscr{A}_2$ and $\mathscr{A}_3$ in a robust RANSAC procedure using the covariance weighted residuals of the points to identify outliers.

We directly obtain the residual for every inlier point by computing its distance to the plane in the direction of the normal directions where each point belongs either to $\mathscr{A}_1$, $\mathscr{A}_2$ or $\mathscr{A}_3$. Using all transformed points from all 30 stereo pairs with their angle ϕ from the optical axis of the camera, we estimate the best plane and update the variance factors $\widehat{\sigma}_1$ and $\widehat{\sigma}_2$ according to Eq. (5.22). This is done iteratively, updating the estimated variance factors and scaling the covariance matrices Σ_{pp} according to the point specific angle ϕ.

We use the exponent $p = 8$ in Eq. (5.22) as this model describes the robust determined variances of the residuals over ϕ best. The dots on the red line in Figure 5.4 indicate the obtained standard deviations using a robust version of variance factor estimation. For this, we determine the variance of the residuals for narrow ranges of ϕ by partitioning the set of all ϕ_n in 30 equally sized bins. For each bin, we use $\widehat{\sigma}_{l'} = 1.48\,\text{MAD}$ with the median absolute difference (MAD) of the residuals to obtain a robust estimate for the standard deviation, see Koch (1999) for details. The blue line in Figure 5.4 shows the estimated functional model of $\widehat{\sigma}_{l'}$ in Eq. (5.22) in dependency of ϕ with $p = 8$, which is close to the 30 determined standard deviations.

Figure 5.5 shows the estimated standard deviation $\widehat{\sigma}_{l'}$ after convergence in dependency of ϕ using the disparities obtained with ELAS (as in Figure 5.4) and SGM. Both curves have the similar shape and the difference amounts to about 0.2 pixel. As this figure shows, measurements having an angle ϕ less than 40° from the optical axis have the highest and nearly constant precision of 0.3 and 0.5 pixel. Beyond 40° the precision degrades revealing the substantially smaller precision of the disparities towards the image borders. By knowing this function, we can now exploit this information in the improved stochastic model.

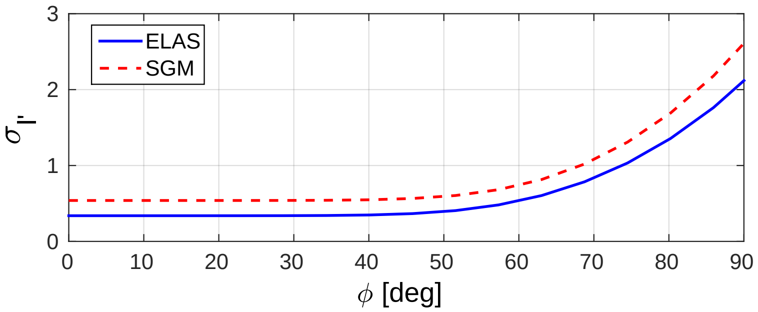


Figure 5.5: Estimated standard deviation $\widehat{\sigma}_{l'}$ over angle ϕ using disparities from ELAS (solid, blue) and SGM (dashed, red).

5.6.2 Orthogonality of Planes

Table 5.1 shows the empirically derived mean and standard deviation of all derived 30 angles. The improved stochastic model that considers the influence of ϕ on the precision of the 3D points leads to smaller deviations from orthogonality and is therefore closer to reality. The empirically derived standard deviations $\widehat{\sigma}_w$ of the angles between the planes confirm a higher precision. The theoretic standard deviation σ_ω that can be obtained given our model is on average 0.424 (ELAS) and 0.676 (SGM) times smaller using the estimated variance factors we obtained in practice. The quotient $\widehat{\sigma}_\omega/\sigma_\omega$ is throughout in the range of $\sqrt{2/30}$ around one, i.e.

$$1 - \sqrt{2/30} < \widehat{\sigma}_\omega/\sigma_\omega < 1 + \sqrt{2/30}\,, \tag{5.27}$$

hence the proposed improved stochastic model of the observation process complies with the empirical results.

5.6.3 Application Examples

Finally, we want to illustrate that the described approach is able to build dense 3D point clouds in real world situations. We show results from an indoor and an outdoor scene.

Figure 5.6 shows the point cloud derived from a stereo image taken in an office with the fisheye stereo camera described before. The disparity information is obtained with ELAS on the epipolar rectified images. The color of each point corresponds in the left

		$\angle(\mathcal{A}_1, \mathcal{A}_2)$	$\angle(\mathcal{A}_1, \mathcal{A}_3)$	$\angle(\mathcal{A}_2, \mathcal{A}_3)$
ELAS	classical	89.69° ± 1.49°	90.23° ± 0.89°	89.74° ± 0.96°
	improved	89.93° ± 0.63°	90.02° ± 0.65°	90.04° ± 0.36°
SGM	classical	89.95° ± 1.29°	89.92° ± 1.19°	89.80° ± 0.58°
	improved	89.93° ± 0.76°	89.94° ± 0.84°	89.87° ± 0.30°

Table 5.1: Empirically derived mean and standard deviation $\widehat{\sigma}_\omega$ of the 30 estimated angles ω between two orthogonal planes using the disparity information from ELAS/SGM and the classical or the improved stochastic model.

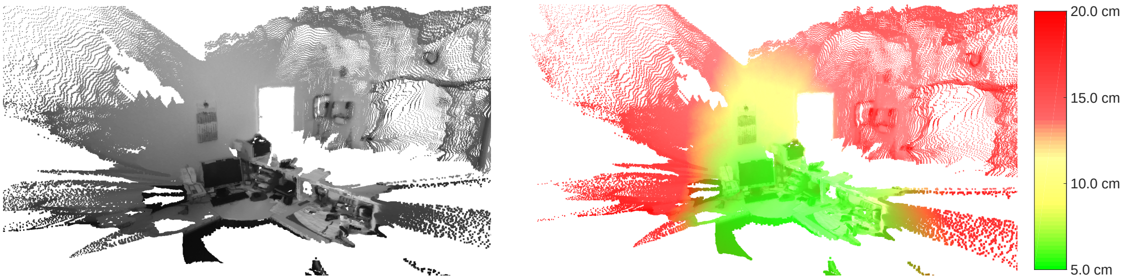


Figure 5.6: Left: Point cloud obtained with disparity information from ELAS. The intensity values correspond to the image content of the left image. Right: Color according to the position accuracy of 3D point, which ranges from 5 cm to 20 cm (green high, red low accuracy).

image to the recorded pixel intensity. In the right image, the intensities are overlaid with the theoretical precision obtained with the estimated variance model. The color spectrum goes from green, for points with highest precisions of about 5 cm, to yellow and red, for points with lowest precision up to 20 cm. Highest precision is achieved for points on the desk as the angles γ_ψ of intersecting rays from both cameras are high and the angle ϕ is small in the center of the image, see Figure 5.6. Points on the wall behind the desk have smaller disparity angles γ_ψ thus less precision (yellow). The precision decreases with increasing angle ϕ and leads to more noisy 3D points more distant to the camera axis (red).

In the last example, we compare the point cloud of an agricultural surface obtained with a fisheye stereo image taken from our copter with a reference point cloud. To compare both point clouds, a rigid body transformation was estimated using corresponding 3D points. The reference point cloud has a point accuracy of about 1 cm and was obtained by bundle adjustment and a subsequent densification using high resolution images taken with high-end equipment. The stereo camera is tilted by 45° towards the ground and the dense depth information from the fisheye stereo image pair is shown in Figure 5.7. It depicts the colored reference point cloud overlayed with the fisheye cloud. The different color encoding shows the absolute error for each point and the histogram illustrates the error distribution. As can be seen, the quality of dense stereo information decays away from the optical axis as the stochastic model predicts. Areas with high errors also have a high theoretical uncertainty.

5.6.4 Remarks

The processing of a fisheye stereo image pair, which includes the rectification, disparity determination and mapping of the 3D points, takes 150 ms per image pair in our ROS implementation using the default robotics parameters in ELAS and thus enables real-time applications. Thus, we can process stereo images with 6-7 Hz, which is suitable for online operation in several application scenarios.

Our experiments suggest that in combination with fisheye epipolar rectification, ELAS

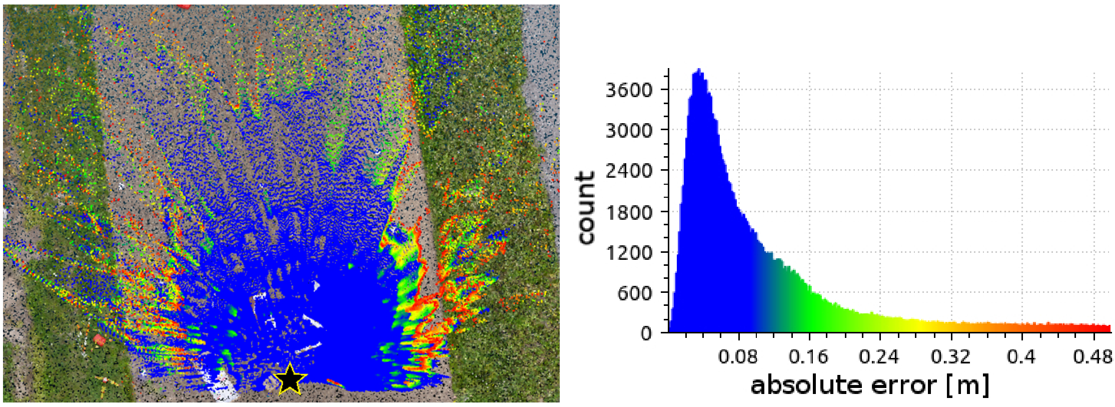


Figure 5.7: Point cloud obtained from a single 752×480 pixel fisheye image pair overlayed with a reference point cloud. The histogram illustrates the distribution of the absolute distances between nearest neighbors encoded with different colors. The black star marks the position of the copter at the time of exposure.

and SGM can both be used for dense fisheye stereo and both methods perform similarly. The precision of the 3D points decreases with angle ϕ. For $\phi > 40°$, the precision drops substantially and leads to more noisy 3D points. This information can be exploited within the observation model. In our experiments, the improved model for the noise in the observations yields a better estimate than the standard model. The theoretic standard deviation σ_ω is on average between 0.42 and 0.68 times smaller than the ones obtained experimentally.

5.7 Conclusion

In this chapter, we analyzed an approach to exploit existing dense stereo methods with wide-angle and fisheye cameras that have a field of view of more than 180°. By conducting fisheye calibration and epipolar rectification beforehand, we can use existing state-of-the-art dense stereo methods as a black box. We thoroughly investigated the accuracy potential of such a fisheye stereo approach and derived an estimate of the uncertainty of the obtained 3D point cloud.

We furthermore generalized the canonical stochastic model for sensor points based on an empirical analysis. The empirical analysis is based on image pairs of a calibrated fisheye stereo camera system and two state-of-the-art algorithms for dense stereo applied to adequately rectified image pairs from fisheye stereo cameras. We showed (1) that adequately rectified fisheye image pairs and dense methods provide dense 3D point clouds at 6 - 7 Hz, (2) that the uncertainty of image points depends on their angular distance from the center of symmetry, (3) how to estimate the parameters of a variance component model, and (4) how the improved stochastic model for the observations influences the accuracy of the 3D points. Note that our method is not limited to a specific fisheye stereo camera system.

6 Discussion

In its first section, this chapter draws a conclusion on the results of this thesis. In its second section, we point out remaining issues for future research directions.

6.1 Conclusion

In this thesis, we focused on the problem of modeling bundle adjustment for omnidirectional and multi-view cameras and presented solutions to problems in the context of visual odometry with an omnidirectional multi-camera system.

In Chap. 3, we started with modeling a more general approach to bundle adjustment. We proposed a bundle adjustment for omnidirectional multi-view camera systems with synchronized times of exposure. The bundle adjustment enables an efficient maximum likelihood estimation and includes image and scene points at infinity which classical approaches are not capable of. As a result, we obtain an increased precision for the estimated camera rotations when using our rigorous estimation procedure which includes far points. Our experiments show that for a variety of multi-camera systems the proposed bundle adjustment can be used for system self-calibration to obtain maximum likelihood estimates for the relative camera poses of the single-view cameras.

In Chap. 4, we considered the problem of real-time visual odometry on a lightweight UAV equipped with a multi-camera system with fisheye cameras which is based on the bundle adjustment developed in the preceding chapter. We presented an effective system for online pose and simultaneous map estimation designed for light-weight UAVs. Our system performs a keyframe-based bundle adjustment in an initially unknown scene based on tracked image features and optionally IMU and GPS observations to incrementally refine an extended map. Incremental bundle adjustment is performed by using the iSAM2 algorithm for sparse nonlinear incremental optimization in combination with our bundle adjustment approach. Experiments show the high potential of the incremental bundle adjustment w.r.t. time requirements and optimality and that a high accuracy level in position can be obtained, which is in the order of RTK GPS.

In addition to that, we presented an effective bundle adjustment solution exploiting RTK-GPS carrier phase observations, IMU data and visual data from feature tracking in an incremental fashion. Our evaluation shows that the overall system yields a robust pose estimate at high frequencies and can handle underconstrained GPS situations effectively. By comparing our estimated georeferenced map obtained with bundle adjustment with a georeferenced terrestrial laser scan we obtain absolute deviations which have a median of less than 1 cm.

Finally, in Chap. 5, we analyzed an approach to exploit existing dense stereo methods with fisheye cameras that have a field of view of more than 180°. We showed that existing state-of-the-art dense stereo methods can be used as a black box when conducting fisheye calibration and epipolar rectification beforehand. We thoroughly investigated the accuracy potential of such a fisheye stereo approach and derived an estimate of the uncertainty of the obtained 3D point cloud.

We furthermore generalized the canonical stochastic model for image points based on an empirical analysis. The empirical analysis is based on image pairs of a calibrated fisheye stereo camera system and two state-of-the-art algorithms for dense stereo applied to adequately rectified image pairs from fisheye stereo cameras. Our investigations show

- that the uncertainty of image points depends on their angular distance from the center of symmetry,
- how to estimate the parameters of a variance component model, and
- how the improved stochastic model for the observations influences the accuracy of the 3D points.

Our method is not limited to a specific fisheye stereo camera system.

The contributions of this thesis are solutions to various aspects in the context of visual odometry with omnidirectional camera systems especially with fisheye cameras. In summary, the approaches presented in this thesis allow us to answer the following questions:

- How to model bundle adjustment to allow for omnidirectional multi-camera systems and to integrate points at infinity?
- How to design effective online pose and simultaneous map estimation for light- weight UAVs?
- How to integrate GPS double difference information into bundle adjustment for accurate and georeferenced pose and map estimation?
- How to model the precision of image points of fisheye images depending on the angular distance from the optical axis?

This thesis lays the foundations for answering these questions.

6.2 Future Work

To conclude this thesis, we would like to highlight the following themes for future research.

6.2.1 Integration of Inequality Constraints for Far Points

Our approach to bundle adjustment proposed in Chap. 3 supports omnidirectional cameras, multi-camera systems and the estimation of points at infinity. However, the iterative estimation of far points may lead to diverging camera rays. Diverging camera rays occur due to the uncertainty of the estimated camera orientations and the uncertainty of the observed camera rays of far points that have small intersection angles. To avoid diverging camera rays, one could formulate inequality constraints forcing camera rays to intersect in front of the cameras. Such constraints can be formulated in quadratic programs, which can be solved e.g. with interior point or active set methods. Jimenez (2016) recently integrated the active set method into the graph-based optimization frame work GTSAM and shows that large quadratic programs can be solved. The integration of inequality constraints has the potential to further improve the quality of camera rotation estimation in the presence of small intersection angles.

6.2.2 Modeling of Unstable Multi-camera Systems

Additionally, in Chap. 4 we did not investigate the effect of vibrations on the camera system due to the motor engines of the UAV. We assumed the camera system to be perfectly stable, which appeared to be a good approximation due to the encouraging results presented in this thesis. However, variations between the relative orientations of the cameras in the multi-camera system could be considered with uncertain prior information or by employing approaches based on a physical model.

6.2.3 Deep Learning Approaches

Deep Learning approaches have already become the dominant approach to achieve state of the art results in many vision problems. However, for the SLAM and visual odometry problem or related 3D geometry problems there is very limited work yet.

More recently, the work of Wang et al. (2017) has shown first success on estimating visual odometry from video using deep learning in an end-to-end fashion without the need of any module of a classical visual odometry pipeline in a pure black box. They employ a Recurrent Convolutional Neural Network which is trained via a supervised known pose signal and shows competitive performance to state-of-the art methods. Zhou et al. (2017) have shown how to train deep networks end-to-end for monocular camera motion and depth estimation completely unsupervised by using view synthesis as the supervisory signal. Such end-to-end solutions have the great advantage of not requiring careful engineering to make submodules of a classical visual odometry pipeline work flawlessly together in different environments. How accurate such end-to-end approaches can get needs to be investigated in future works.

Additionally, current research shows success in solving subtasks of the SLAM problem with deep neural networks. Classical pipeline stages heavily rely on accurate image correspondences, which is why areas of low texture, complex geometry or occlusions may cause problems. Recent works address these problems by learning end-to-end regression using deep learning with promising results. For example, Han et al. (2015) perform feature matching by training a patch matching system end-to-end, Kendall et al. (2015) train a convolutional neural network to obtain the 6 DoF camera pose from a single image, and Kendall et al. (2017) use deep learning to learn an end-to-end mapping from an image pair to disparity maps and achieve high quality results on traditionally difficult scenes.

Approaches using deep learning are highly promising as they are expected to lead to a lot of progress in this field.

List of Figures

1.1 Dense 3D surface reconstruction from images. 13
1.2 Illustration of the multirotor and its sensor setup. 14
1.3 A synchronized triggered frame set of the UAV's four fisheye cameras. . . . 15

2.1 Illustration of the displaced coordinate systems of the corresponding example. 25
2.2 Illustration of different mapping principles of central projection cameras. . . 27
2.3 Relation between a camera ray direction and a projected image point. . . . 28
2.4 The two test-fields for camera calibration. 31
2.5 A small optimization problem represented as factor graph, Bayes net and Bayes tree. 40

3.1 Four camera designs for multi-camera systems. 42
3.2 Images of a multi-camera system with fisheye cameras. 43
3.3 Local flight at Woodruf Ave with far point towards west end. 44
3.4 Inverse depth representation. 44
3.5 A two-camera system with fisheye cameras shown at two exposure times observing points observing near and far points. 47
3.6 Homogeneous image coordinate vector and ray direction with different sign. 52
3.7 Relation between sensor point, viewing direction and viewing ray in the equi-distant projection model. 53
3.8 Illustration of the uncertainty of a point represented in the tangent space on the unit sphere and the estimation update in the tangent space. 57
3.9 Simulation of a moving multi-camera system with loop closing. 62
3.10 Sample images of the stereo camera dataset. 63
3.11 Illustration of the estimated scene points and poses of the FinePix Real 3D W1 dataset. 63
3.12 Sample images of the Ladybug 3 dataset. 64
3.13 Illustration of the results of the multi-camera dataset evaluation. 65
3.14 Illustration of the results of the urban drive dataset evaluation. 67
3.15 Multi-camera system consisting of five overlapping perspective camera views. 68

3.16 Illustration of the estimated scene points and poses of the reference camera and of the estimated relative poses of the multi-camera system. 68

3.17 The Ladybug 3 mounted on a robot executing a circular movement. 69

3.18 Comparison of the estimated and manufactured given relative poses. 70

3.19 Illustration of the UAV. One stereo pair of the UAV is looking forward and one backwards, which provides a wide field of view. 70

4.1 Synchronized triggered frame set of the four fisheye cameras. 77

4.2 Fisheye stereo pair with curved epipolar line. 78

4.3 llustration of the keyframe-based bundle adjustment. 80

4.4 Illustration of the factor graph of the keyframe-based bundle adjustment with camera information only. 81

4.5 The RTK-GPS/IMU state estimation board. 82

4.6 Illustration of the factor graph of the keyframe-based bundle adjustment which integrates GPS and IMU information. 84

4.7 Double differences from distances between known positions of GPS satellites and master and rover position. 84

4.8 Illustration of the required time to incrementally process a keyframe, its dependency on the number of new factors and the number of affected variables. 86

4.9 Root mean square error of extracted image points for each keyframe. 87

4.10 Deviations between the estimated rotation angles and translations of BACS and iSAM2 on all keyframes. 87

4.11 The theoretical precision of the positions from GPS, from pure visual odometry and from visual odometry which integrates GPS. 88

4.12 The deviations between the keyframe positions from visual odometry and the GPS coordinates. 88

4.13 Cumulative histogram of the track lengths of a flight with four fisheye cameras. 89

4.14 Trajectory of the UAV flight overlayed with a georeferenced 3D model. . . . 89

4.15 Theoretical standard deviation of estimated pose parameters at keyframes. 90

4.16 Residuals between incrementally estimated positions of keyframes for the GPS double differences. 91

4.17 Illustration of three estimated trajectories showing the benefit of operating on raw GPS DD measurements. 92

4.18 Measured intensities of a dense laser scan in panoramic grayscale image. . . 92

4.19 Comparison of mapped point cloud with georeferenced terrestrial laser scan. 93

5.1 UAV equipped with fisheye stereo cameras. 96

5.2 The projection of epipolar planes inside the image rows. 101

5.3 Stereo camera with fisheye lenses, highly textured and mutually orthogonal planes and stereo image pair before and after epipolar rectification. 106
5.4 Robust estimates for improved stochastic model. 107
5.5 Estimated stochastic model using disparities from ELAS and SGM. 107
5.6 Point cloud obtained with disparity information from ELAS. 109
5.7 Point cloud obtained from a single fisheye image pair. 109

Bibliography

Abraham, S. (1999). *Kamera-Kalibrierung und metrische Auswertung monokularer Bildfolgen.* PhD thesis, University of Bonn, Institute of Geodesy and Geoinformation.

Abraham, S. and Förstner, W. (2005). Fish-eye-stereo Calibration and Epipolar Rectification. *ISPRS Journal of Photogrammetry and Remote Sensing (JPRS)*, 59(5):278–288.

Abraham, S. and Hau, T. (1997). Towards Autonomous High-Precision Calibration of Digital Cameras. In *Proc. of SPIE Videometrics*, volume 3174, pages 82–93.

Agarwal, S., Furukawa, Y., Snavely, N., Simon, I., Curless, B., Seitz, S., and Szeliski, R. (2011). Building Rome in a Day. *Communications of the ACM (CACM)*, 54(10).

Agarwal, S., Mierle, K., and Others (2018). Ceres Solver. `http://ceres-solver.org`.

Arfaoui, A. and Thibault, S. (2015). Mathematical model for hybrid and panoramic stereovision systems: panoramic to rectilinear conversion model. *Applied Optics*, 54(21):6534–6542.

Aubry, M., Kolev, K., Goldluecke, B., and Cremers, D. (2011). Decoupling Photometry and Geometry in Dense Variational Camera Calibration. In *Proc. of the IEEE Int. Conf. on Computer Vision (ICCV)*, pages 1411–1418.

Baarda, W. (1967). *Statistical Concepts in Geodesy*, volume 2 of *Publications on Geodesy, New Series.* Netherlands Geodetic Commission, 4th edition.

Bartoli, A. (2002). On the Non-linear Optimization of Projective Motion Using Minimal Parameters. In *Proc. of the European Conf. on Computer Vision (ECCV)*, volume 2351 of *Lecture Notes in Computer Science (LNCS)*, pages 340–354.

Bäumker, M., Przybilla, H.-J., and Zurhorst, A. (2013). Enhancements in UAV flight control and sensor orientation. In *ISPRS Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XL-1/W2, pages 33–38.

Beekmans, C., Schneider, J., Läbe, T., Lennefer, M., Stachniss, C., and Simmer, C. (2016). Cloud Photogrammetry with Dense Stereo for Fisheye Cameras. *Atmospheric Chemistry and Physics (ACP)*, 16(22):14231–14248.

Bennewitz, M., Stachniss, C., Burgard, W., and Behnke, S. (2006). Metric Localization with Scale-Invariant Visual Features using a Single Perspective Camera. In *Proc. of the European Robotics Symposium (EUROS)*, pages 143–157.

Bouguet, J. (2000). Pyramidal Implementation of the Lucas Kanade Feature Tracker. Technical report, Intel Corporation, Microprocessor Research Labs.

Bradski, G. and Kaehler, A. (2008). *Learning OpenCV.* O'Reilly, 1st edition.

Brown, D. (1958). *A Solution to the General Problem of Multiple Station Analytical Stereotriangulation.* RCA Data Reducation Technical Report No. 43. D. Brown Associates, Incorporated.

Brown, D. (1971). Close-Range Camera Calibration. *Photogrammetric Engineering (PE)*, 37(8):855–866.

Carrera, G., Angeli, A., and Davison, A. (2011). SLAM-based Automatic Extrinsic Calibration of a Multi-Camera Rig. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, pages 2652–2659.

Chang, X.-W., Yang, X., and Zhou, T. (2005). MLAMBDA: A modified LAMBDA method for integer least-squares estimation. *Journal of Geodesy*, 79(9):552–565.

Choi, K. and Lee, I. (2012). A Sequential Aerial Triangulation Algorithm for Real-time Georeferencing of Image Sequences Acquired by an Airborne Multi-Sensor System. *Remote Sensing*, 5(1):57–82.

Civera, J., Davison, A., and Montiel, J. (2008). Inverse Depth Parametrization for Monocular SLAM. *IEEE Trans. on Robotics (TRO)*, 24(5):932–945.

Davis, T., Gilbert, J., Larimore, S., and Ng, E. (2004). A Column Approximate Minimum Degree Ordering Algorithm. *ACM Trans. on Mathematical Software (TOMS)*, 30(3):353–376.

Davison, A. (2003). Real-Time Simultaneous Localisation and Mapping with a Single Camera. In *Proc. of the IEEE Int. Conf. on Computer Vision (ICCV)*, volume 2, pages 1403–1410.

Davison, A., Cid, Y. G., and Kita, N. (2004). Real-Time 3D SLAM with Wide-Angle Vision. In *Proc. of the IFAC Symposium on Intelligent Autonomous Vehicles (IAV)*, volume 37, pages 868–873.

Dellaert, F. (2012). Factor Graphs and GTSAM: A Hands-on Introduction. Technical Report GT-RIM-CP&R-2012-002., Georgia Institute of Technology.

Eisenbeiss, H., Lambers, K., Sauerbier, M., and Li, Z. (2005). Photogrammetric documentation of an archaeological site (Palpa, Peru) using an autonomous model helicopter. In *ISPRS Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXVI-5/C34, pages 238–243.

Eling, C., Heinz, E., Klingbeil, L., and Kuhlmann, H. (2014). Cycle Slip Detection in the context of RTK GPS positioning of lightweight UAVs. In *Proc. of the Int. Conf. on Machine Control and Guidance (MCG)*, pages 148–155.

Eling, C., Klingbeil, L., and Kuhlmann, H. (2015). Real-Time Single-Frequency GPS/MEMS-IMU Attitude Determination of Lightweight UAVs. *IEEE Sensors Journal*, 15(10):26212–26235.

Ellum, C. (2004). Integration of Raw GPS Measurements into a Bundle Adjustment. In *ISPRS Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXV-B4, pages 933–938.

Engel, J., Koltun, V., and Cremers, D. (2018). Direct Sparse Odometry. *IEEE Trans. on Pattern Analalysis and Machine Intelligence (TPAMI)*, 40(3):611–625.

Engel, J., Sturm, J., and Cremers, D. (2013). Semi-Dense Visual Odometry for a Monocular Camera. In *Proc. of the IEEE Int. Conf. on Computer Vision (ICCV)*, pages 1449–1456.

Engels, C., Stewénius, H., and Nistér, D. (2006). Bundle Adjustment Rules. In *ISPRS Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXVI Part 3, pages 266–271.

Esparza, J., Helmle, H., and Jähne, B. (2014). Wide Base Stereo with Fisheye Optics: A Robust Approach for 3D Reconstruction in Driving Assistance. In *Proc. of the German Conf. on Pattern Recognition (GCPR)*, volume 8753 of *Lecture Notes in Computer Science (LNCS)*, pages 342–353.

Forster, C., Pizzoli, M., and Scaramuzza, D. (2013). Air-Ground Localization and Map Augmentation Using Monocular Dense Reconstruction. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 3971–3978.

Förstner, W. (2012). Minimal Representations for Testing and Estimation in Projective Spaces. *Photogrammetrie – Fernerkundung – Geoinformation (PFG)*, 3:209–220.

Förstner, W. (2017). Towards Real Time Visual Odometry and Mapping with UAVs. Tutorial at the Int. Conf. on Unmanned Aerial Vehicles in Geomatics (UAV-g).

Förstner, W. and Wrobel, B. (2016). *Photogrammetric Computer Vision – Statistics, Geometry, Orientation and Reconstruction*, volume 11 of *Geometry and Computing*. Springer, 1st edition.

Frahm, J., Köser, K., and Koch, R. (2004). Pose Estimation for Multi-camera Systems. In *Proc. of the Annual Symposion of the German Association for Pattern Recognition (DAGM)*, volume 3175 of *Lecture Notes in Computer Science (LNCS)*, pages 286–293.

Franke, U., Pfeiffer, D., C.Rabe, Knoeppel, C., Enzweiler, M., Stein, F., and Herrtwich, R. (2013). Making Bertha See. In *Proc. of the IEEE Int. Conf. on Computer Vision Workshops (ICCVW)*, pages 214–221.

Fraser, C. (2001). Photogrammetric Camera Component Calibration: A Review of Analytical Techniques. In Gruen, A. and Huang, T., editors, *Calibration and Orientation of Cameras in Computer Vision*, Springer Series in Information Sciences, pages 95–121. Springer.

Fu, Q., Quan, Q., and Cai, K.-Y. (2014). Calibration of Multiple Fish-Eye Cameras Using a Wand. *IET Computer Vision*, 9(3):378–389.

Gallup, D., Frahm, J., Mordohai, P., Yang, Q., and Pollefeys, M. (2007). Real-Time Plane-Sweeping Stereo with Multiple Sweeping Directions. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, number 1-8.

Gao, W. and Shen, S. (2017). Dual-Fisheye Omnidirectional Stereo. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 6715–6722.

Geiger, A., Roser, M., and Urtasun, R. (2010). Efficient Large-Scale Stereo Matching. In *Proc. of the Asian Conf. on Computer Vision (ACCV)*, volume 6492 of *Lecture Notes in Computer Science (LNCS)*, pages 25–38.

Golub, G. and Loan, C. V. (1996). *Matrix Computations*. The Johns Hopkins University Press, 3rd edition.

Grün, A. (1984). Algorithmic Aspects in on-line Triangulation. In *ISPRS Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXV-A3, pages 342–362.

Grün, A. (1985). Algorithmic Aspects in On-Line Triangulation. *Photogrammetric Engineering and Remote Sensing (PE&RS)*, 51(4):419–436.

Grün, A. (1987). Towards real-time Photogrammetry. In *Proc. of the Photogrammetric Week (Phowo)*.

Hampel, F., Ronchetti, E., Rousseeuw, P., and Stahel, W. (1986). *Robust Statistics: The Approach Based on Influence Functions*. Johan Wiley & Sons.

Han, X., Leung, T., Jia, Y., Sukthankar, R., and Berg, A. (2015). MatchNet: Unifying Feature and Metric Learning for Patch-based Matching. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 3279–3286.

Häne, C. (2016). *Semantic 3D Modeling from Images with Geometric Priors*. PhD thesis, ETH Zurich.

Harmat, A., Trentini, M., and Sharf, I. (2015). Multi-Camera Tracking and Mapping for Unmanned Aerial Vehicles in Unstructured Environments. *Journal of Intelligent and Robotic Systems (JIRS)*, 78(2):291–317.

Hartley, R. and Zisserman, A. (2004). *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2nd edition.

Heller, J. and Pajdla, T. (2009). Stereographic Rectification of Omnidirectional Stereo Pairs. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 1414–1421.

Hemayed, E. (2003). A Survey on Camera Self-calibration. In *Proc. of the IEEE Conf. on Advanced Video and Signal Based Surveillance (AVSS)*, pages 351–357.

Heng, L., Bürki, M., Lee, G., Furgale, P., Siegwart, R., and Pollefeys, M. (2014). Infrastructure-Based Calibration of a Multi-Camera Rig. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*.

Herrera, P., Pajares, G., Guijarro, M., Ruz, J., and Cruz, J. (2011). A Stereovision Matching Strategy for Images Captured with Fish-Eye Lenses in Forest Environments. *IEEE Sensors Journal*, 11:1756–1783.

Heuel, S. (2004). *Uncertain Projective Geometry: Statistical Reasoning for Polyhedral Object Reconstruction*, volume 3008 of *Lecture Notes in Computer Science (LNCS)*. Springer.

Hirschmüller, H. (2008). Stereo Processing by Semi-Global Matching and Mutual Information. *IEEE Trans. on Pattern Analalysis and Machine Intelligence (TPAMI)*, 30(2):328–341.

Hirschmüller, H. (2011). Semi-Global Matching – Motivation, Developments and Applications. In *Proc. of the Photogrammetric Week (Phowo)*, pages 173–184.

Hofmann-Wellenhof, B., Lichtenegger, H., and Wasle, E. (2008). *GNSS – Global Navigation Satellite Systems*. Springer Vienna, 1st edition.

Huang, K. and Stachniss, C. (2017). Extrinsic Multi-Sensor Calibration For Mobile Robots Using the Gauss-Helmert Model. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 1490–1496.

Huber, P. (1981). *Robust Statistics*. John Wiley.

Indelman, V. and Dellaert, F. (2015). Incremental Light Bundle Adjustment: Probabilistic Analysis and Application to Robotic Navigation. In Sun, Y., Behal, A., and Chung, C.-K. R., editors, *New Development in Robot Vision*, volume 23 of *Cognitive Systems Monographs*, chapter 7, pages 111–136. Springer.

Ishiguro, H., Yamamato, M., and Tsuji, S. (1992). Omni-Directional Stereo. *IEEE Trans. on Pattern Analalysis and Machine Intelligence (TPAMI)*, 14(2):257–262.

Jimenez, I. (2016). *A Factor Graph Approach to Constrained Optimization*. PhD thesis, Georgia Institute of Technology.

Julier, S. and Uhlmann, J. (2001). A Counter Example to the Theory of Simultaneous Localization and Map Building. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, volume 4, pages 4238–4243.

Kaess, M., Ila, V., Roberts, R., and Dellaert, F. (2010). The Bayes Tree: An Algorithmic Foundation for Probabilistic Robot Mapping. In *Int. Workshop on the Algorithmic Foundations of Robotics (WAFR)*, volume 68 of *Springer Tracts in Advanced Robotics*, pages 157–173.

Kaess, M., Johannsson, H., Roberts, R., Ila, V., Leonard, J., and Dellaert, F. (2012). iSAM2: Incremental Smoothing and Mapping Using the Bayes Tree. *Int. Journal of Robotics Research (IJRR)*, 31(2):217–236.

Kaess, M., Ranganathan, A., and Dellaert, F. (2008). iSAM: Incremental Smoothing and Mapping. *IEEE Trans. on Robotics (TRO)*, 24(6):1365–1378.

Kanatani, K. (1996). *Statistical Optimization for Geometric Computation: Theory and Practice*. Elsevier Science, 1st edition.

Kendall, A., Grimes, M., and Cipolla, R. (2015). PoseNet: A Convolutional Network for Real-Time 6-DOF Camera Relocalization. In *Proc. of the IEEE Int. Conf. on Computer Vision (ICCV)*, pages 2938–2946.

Kendall, A., Martirosyan, H., Dagupta, S., and Henry, P. (2017). End-to-End Learning of Geometry and Context for Deep Stereo Regression. In *Proc. of the IEEE Int. Conf. on Computer Vision (ICCV)*, pages 66–75.

Kim, J. (2010). *Camera Motion Estimation for Multi-Camera Systems*. PhD thesis, School of Engineering, ANU College of Engineering and Computer Science, The Australian National University.

Kingston, D. and Beard, R. (2004). Real-Time Attitude and Position Estimation for Small UAVs Using Low-Cost Sensors. In *Proc. of AIAA 3rd "Unmanned Unlimited" Technical Conference, Workshop and Exhibit*, volume 1, pages 489–497.

Kita, N. (2011a). Dense 3D Measurement of the Near Surroundings by Fisheye Stereo. In *Proc. of the IAPR Conf. on Machine Vision Applications (MVA)*, pages 148–151.

Kita, N. (2011b). Direct floor height measurement for biped walking robot by fisheye stereo. In *IEEE Int. Conf. on Humanoid Robots*, pages 187–192.

Klein, G. and Murray, D. (2007). Parallel Tracking and Mapping for Small AR Workspaces. In *Proc. of the Int. Symposium on Mixed and Augmented Reality (ISMAR)*, pages 1–10.

Klingbeil, L., Nieuwenhuisen, M., Schneider, J., Eling, C., Dröschel, D., Holz, D., Läbe, T., Förstner, W., Behnke, S., and Kuhlmann, H. (2014). Towards Autonomous Navigation of an UAV-based Mobile Mapping System. In *Proc. of the Int. Conf. on Machine Control and Guidance (MCG)*, pages 136–147.

Klingner, B., Martin, D., and Roseborough, J. (2013). Street View Motion-from-Structure-from-Motion. In *Proc. of the IEEE Int. Conf. on Computer Vision (ICCV)*, pages 953–960.

Knapitsch, A., Park, J., Zhou, Q., and Koltun, V. (2017). Tanks and Temples: Benchmarking Large-Scale Scene Reconstruction. *ACM Trans. on Graphics (TOG)*, 36(4).

Koch, K. (1999). *Parameter Estimation and Hypothesis Testing in Linear Models.* Springer, 2nd edition.

Kraus, K., Hartley, I., and Kyle, S. (2011). *Photogrammetry. Geometry from Images and Laser Scans.* De Gruyter, 2nd edition.

Kschischang, F., Frey, B., and Loeliger, H. (2001). Factor Graphs and the Sum-Product Algorithm. *IEEE Trans. on Information Theory*, 47(2):498–519.

Kümmerle, R., Grisetti, G., Strasdat, H., Konolige, K., and Burgard, W. (2011). G2o: A General Framework for Graph Optimization. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, pages 3607–3613.

Läbe, T. and Förstner, W. (2006). Automatic Relative Orientation of Images. In *Proc. of the 5th Turkish-German Joint Geodetic Days.*

Li, L., Yu, X., Zhang, S., Zhao, X., and Zhang, L. (2017). 3D Cost Aggregation with Multiple Minimum Spanning Trees for Stereo Matching. *Applied Optics*, 56(12):3411–3420.

Li, L., Zhang, S., Yu, X., and Zhang, L. (2018). PMSC: PatchMatch-Based Superpixel Cut for Accurate Stereo Matching. *IEEE Trans. on Circuits and Systems for Video Technology (TCSVT)*, 28(3):679–692.

Lourakis, M. and Argyros, A. (2009). SBA: A Software Package for Generic Sparse Bundle Adjustment. *ACM Trans. on Mathematical Software (TOMS)*, 36(1):1–30.

Lu, F. and Milios, E. (1997). Globally Consistent Range Scan Alignment for Environment Mapping. *Autonomous Robots*, 4(4):333–349.

Ly, D., Demonceaux, C., Vasseur, P., and Pégard, C. (2014). Extrinsic calibration of heterogeneous cameras by line images. *Proc. of the IAPR Conf. on Machine Vision Applications (MVA)*, 25(6):1601–1614.

Maier, D., Stachniss, C., and Bennewitz, M. (2013). Vision-Based Humanoid Navigation Using Self-Supervised Obstacle Detection. *Int. Journal of Humanoid Robotics (IJHR)*, 10(2):1263–1269.

Mei, C. and Rives, P. (2007). Single View Point Omnidirectional Camera Calibration from Planar Grids. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, pages 3945–3950.

Meidow, J., Beder, C., and Förstner, W. (2009). Reasoning with Uncertain Points, Straight Lines, and Straight Line Segments in 2D. *ISPRS Journal of Photogrammetry and Remote Sensing (JPRS)*, 64(2):125–139.

Merz, T. and Kendoul, F. (2011). Beyond visual range obstacle avoidance and infrastructure inspection by an autonomous helicopter. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 4953–4960.

Moreau, J., Ambellouis, S., and Ruichek, Y. (2013). Equisolid Fisheye Stereovision Calibration and Point Cloud Computation. In *ISPRS Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XL-7/W2, pages 167–172.

Mostafa, M. and Schwarz, K. (2001). Digital Image Georeferencing from a Multiple Camera System by GPS/INS. *ISPRS Journal of Photogrammetry and Remote Sensing (JPRS)*, 56(1):1–12.

Mouragnon, E., Lhuillier, M., Dhome, M., Dekeyser, F., and Sayd, P. (2009). Generic and Real-time Structure from Motion Using Local Bundle Adjustment. *Elsevier Journal on Image and Vision Computing (IVC)*, 27(8):1178–1193.

Muhle, D., Abraham, S., Heipke, C., and Wiggenhagen, M. (2011). Estimating the Mutual Orientation in a Multi-camera System with a Non Overlapping Field of View. In *Proc. of the ISPRS Conference on Photogrammeric Image Analysis (PIA)*, volume 6952 of *Lecture Notes in Computer Science (LNCS)*, pages 13–24.

Mur-Artal, R., Montiel, J., and Tardós, J. (2015). ORB-SLAM: A Versatile and Accurate Monocular SLAM System. *IEEE Trans. on Robotics (TRO)*, 31(5):1147–1163.

Newcombe, R., Lovegrove, S., and Davison, A. (2011). DTAM: Dense tracking and mapping in real-time. In *Proc. of the IEEE Int. Conf. on Computer Vision (ICCV)*.

Nguyen, T.-T. and Lhuillier, M. (2016). Adding Synchronization and Rolling Shutter in Multi-Camera Bundle Adjustment. In *Proc. of the British Machine Vision Conference (BMVC)*, pages 62.1–62.11.

Nieuwenhuisen, M., Dröschel, D., Schneider, J., Holz, D., Läbe, T., and Behnke, S. (2013). Multimodal Obstacle Detection and Collision Avoidance for Micro Aerial Vehicles. In

Proc. of the European Conf. on Mobile Robotics (ECMR), pages 7–12.

Nistér, D., Naroditsky, O., and Bergen, J. (2004). Visual Odometry. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 652–659.

Olson, E. (2011). AprilTag: A Robust and Flexible Visual Fiducial System. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, pages 3400–3407.

Olson, E., Leonard, J., and Teller, S. (2006). Fast Iterative Optimization of Pose Graphs with Poor Initial Estimates. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, pages 2262–2269.

Pizzoli, M., Forster, C., and Scaramuzza, D. (2014). REMODE: Probabilistic, Monocular Dense Reconstruction in Real Time. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, pages 2609–2616.

Pollefeys, M., Koch, R., and Gool, L. V. (1999). A Simple and Efficient Rectification Method for General Motion. In *Proc. of the IEEE Int. Conf. on Computer Vision (ICCV)*, volume 1, pages 496–501.

Puig, L., Bermúdez, J., Sturm, P., and Guerrero, J. (2012). Calibration of omnidirectional cameras in practice: A comparison of methods. *Journal of Computer Vision and Image Understanding (CVIU)*, 116(1):120–137.

Rehak, M., Mabillard, R., and Skaloud, J. (2014). A Micro Aerial Vehicle with Precise Position and Attitude Sensors. *Photogrammetrie – Fernerkundung – Geoinformation (PFG)*, 4:239–251.

Rieke, M., Foerster, T., Geipel, J., and Prinz, T. (2011). High-precision positioning and real-time data processing of uav-systems. In *ISPRS Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXVIII-1/C22.

Rudolf, P., Wzorek, M., and Dotherty, P. (2010). Vision-based Pose Estimation for Autonomous Indoor Navigation of Micro-scale Unmanned Aircraft Systems. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, pages 1913–1920.

Savopol, F., Chapman, M., and Boulianne, M. (2000). A Digital Multi CCD Camera System for Near Real-Time Mapping. In *ISPRS Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXIII-B1, pages 266–271.

Scaramuzza, D. (2008). *Omnidirectional Vision: From Calibration to Robot Motion Estimation.* PhD thesis, ETH Zurich.

Scaramuzza, D., Martinelli, A., and Siegwart, R. (2006). A Toolbox for Easily Calibrating Omnidirectional Cameras. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 5695–5701.

Scharstein, D. and Szeliski, R. (2002). A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms. *Int. Journal of Computer Vision (IJCV)*, 47:7–42.

Schmeing, B., Läbe, T., and Förstner, W. (2011). Trajectory Reconstruction Using Long Sequences of Digital Images From an Omnidirectional Camera. In *Proc. of the Annual Conf. of the German Society for Photogrammetry, Remote Sensing and Geoinformation*

(DGPF), pages 1–10.

Schmid, K., Lutz, P., Tomic, T., Mair, E., and Hirschmüller, H. (2014). Autonomous Vision-based Micro Air Vehicle for Indoor and Outdoor Navigation. *Journal of Field Robotics (JFR)*, 31:537–570.

Schneider, J., Eling, C., Klingbeil, L., Kuhlmann, H., Förstner, W., and Stachniss, C. (2016a). Fast and Effective Online Pose Estimation and Mapping for UAVs. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, pages 4784–4791.

Schneider, J. and Förstner, W. (2013). Bundle Adjustment and System Calibration with Points at Infinity for Omnidirectional Camera Systems. *Photogrammetrie – Fernerkundung – Geoinformation (PFG)*, 2013(4):309–321.

Schneider, J. and Förstner, W. (2014). Real-Time Accurate Geo-Localization of a MAV with Omnidirectional Visual Odometry and GPS. In *Computer Vision - ECCV 2014 Workshops*, volume 8925 of *Lecture Notes in Computer Science (LNCS)*, pages 271–282.

Schneider, J., Läbe, T., and Förstner, W. (2013). Incremental Real-time Bundle Adjustment for Multi-camera Systems with Points at Infinity. In *ISPRS Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XL-1/W2, pages 355–360.

Schneider, J., Läbe, T., and Förstner, W. (2014). Real-Time Bundle Adjustment with an Omnidirectional Multi-Camera System and GPS. In *Proc. of the Int. Conf. on Machine Control and Guidance (MCG)*, pages 98–103.

Schneider, J., Schindler, F., and Förstner, W. (2011). Bündelausgleichung für Multikamerasysteme. In *Proc. of the Annual Conf. of the German Society for Photogrammetry, Remote Sensing and Geoinformation (DGPF)*, pages 119–127.

Schneider, J., Schindler, F., Läbe, T., and Förstner, W. (2012). Bundle Adjustment for Multi-camera Systems with Points at Infinity. In *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume I-3, pages 75–80.

Schneider, J., Stachniss, C., and Förstner, W. (2016b). Dichtes Stereo mit Fisheye-Kameras. In *UAV 2016 – Vermessung mit unbemannten Flugsystemen*, volume 82 of *Schriftenreihe des DVW*, pages 247–264. Wißner Verlag.

Schneider, J., Stachniss, C., and Förstner, W. (2016c). On the Accuracy of Dense Fisheye Stereo. *IEEE Robotics and Automation Letters (RA-L)*, 1(1):227–234.

Schneider, J., Stachniss, C., and Förstner, W. (2017). On the Quality and Efficiency of Approximate Solutions to Bundle Adjustment with Epipolar and Trifocal Constraints. In *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume IV-2/W3, pages 81–88.

Seitz, S., Kalai, A., and Shum, H.-Y. (2002). Omnivergent Stereo. *Int. Journal of Computer Vision (IJCV)*, 48(3):159–172.

Shi, J. and Tomasi, C. (1994). Good Features to Track. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 593–600.

Stempfhuber, W. and Buchholz, M. (2011). A precise, low-cost RTK GNSS system for UAV applications. In *ISPRS Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXVIII-1/C22, pages 289–293.

Strasdat, H., Montiel, J., and Davison, A. (2012). Visual SLAM: Why filter? *Elsevier Journal on Image and Vision Computing (IVC)*, 30(2):65–77.

Stühmer, J., Gumhold, S., and Cremers, D. (2010). Real-Time Dense Geometry from a Handheld Camera. In *Proc. of the Annual Symposion of the German Association for Pattern Recognition (DAGM)*, volume 6376 of *Lecture Notes in Computer Science (LNCS)*, pages 11–20.

Swaminathan, R., Grossberg, M., and Nayar, S. (2001). Caustics of Catadioptric Cameras. In *Proc. of the IEEE Int. Conf. on Computer Vision (ICCV)*, volume 2, pages 2–9.

Taketomi, T., Uchiyama, H., and Ikeda, S. (2017). Visual SLAM algorithms: a survey from 2010 to 2016. *IPSJ Transactions on Computer Vision and Applications (CVA)*, 9(16).

Taniai, T., Matsushita, Y., Sato, Y., and Naemura, T. (2017). Continuous 3D Label Stereo Matching using Local Expansion Moves. *IEEE Trans. on Pattern Analalysis and Machine Intelligence (TPAMI)*.

Tarjan, R. and Yannakakis, M. (1984). Simple Linear-Time Algorithms to Test Chordality of Graphs, Test Acyclicity of Hypergraphs, and Selectively Reduce Acyclic Hypergraphs. *SIAM Journal on Computing*, 13(3):566–579.

Tomic, T., Schmid, K., Lutz, P., Dömel, A., Kassecker, M., Mair, E., Grixa, I., Ruess, F., Suppa, M., and Burschka, D. (2012). Toward a Fully Autonomous UAV: Research Platform for Indoor and Outdoor Urban Search and Rescue. *IEEE Robotics and Automation Magazine (RAM)*, 19(3):46–56.

Tommaselli, A. M. G., Marcato Jr, J., Moraes, M. V. A., Silva, S. L. A., and Artero, A. O. (2014). Calibration of Panoramic Cameras with Coded Targets and a 3D Calibration Field. In *ISPRS Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XL-3/W1, pages 137–142.

Triggs, B., McLauchlan, P., Hartley, R., and Fitzgibbon, A. (2000). Bundle Adjustment – A Modern Synthesis. In *Vision Algorithms: Theory and Practice*, volume 1883 of *Lecture Notes in Computer Science (LNCS)*, pages 298–372.

Tsai, R. (1987). A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-shelf TV Cameras and Lenses. *IEEE Journal on Robotics and Automation (RA)*, RA-3(4):323–344.

Urban, S., Wursthorn, S., Leitloff, J., and Hinz, S. (2017). MultiCol Bundle Adjustment: A Generic Method for Pose Estimation, Simultaneous Self-Calibration and Reconstruction for Arbitrary Multi-Camera Systems. *Int. Journal of Computer Vision (IJCV)*, 121(2):234–252.

van der Mark, W. and Gavrila, D. (2006). Real-Time Dense Stereo for Intelligent Vehicles.

IEEE Trans. on Intelligent Transportation Systems, 7(1):38–50.

Žbontar, J. and LeCun, Y. (2016). Stereo Matching by Training a Convolutional Neural Network to Compare Image Patches. *The Journal of Machine Learning Research*, 17(1):2287–2318.

Wang, S., Clark, R., Wen, H., and Trigoni, N. (2017). DeepVO: Towards End-to-End Visual Odometry with Deep Recurrent Convolutional Neural Networks. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, pages 2043–2050.

Wang, W., Xiao, H., Li, W., and Zhang, M. (2015). Enhancement of fish-eye imaging quality based on compressive sensing. *Optik - Int. Journal for Light and Electron Optics*, 126(19):2050–2054.

Wendel, J., Meister, O., Schlaile, C., and Trommer, G. (2006). An integrated GPS/MEMS-IMU navigation system for an autonomous helicopter. *Aerospace Science and Technology*, 10(6):527–533.

Xiang, H. and Tian, L. (2011). Development of a low-cost agricultural remote sensing system based on an autonomous unmanned aerial vehicle (UAV). *Biosystems Engineering*, 108(2):174–190.

Xiong, Y. and Turkowski, K. (1997). Creating image-based VR using a self-calibrating fisheye lens. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 237–243.

Xu, C. and Peng, X. (2014). Fish-eye lens rectification based on equidistant model. In *Proc. of the Int. Conf. on Information Technology and Applications (ITA)*, pages 163–166.

Ying, X. and Hu, Z. (2004). Can We Consider Central Catadioptric Cameras and Fisheye Cameras within a Unified Imaging Model. In *Proc. of the European Conf. on Computer Vision (ECCV)*, volume 3021 of *Lecture Notes in Computer Science (LNCS)*, pages 442–455.

Yoo, C. and Ahn, I. (2003). Low cost GPS/INS sensor fusion system for UAV navigation. In *Proc. of the Digital Avionics Systems Conference (DASC)*, pages 8.A.1–8.1–9.

Zbontar, J. and LeCun, Y. (2015). Stereo Matching by Training a Convolutional Neural Network to Compare Image Patches. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Zhang, Z. (2000). A Flexible New Technique for Camera Calibration. *IEEE Trans. on Pattern Analalysis and Machine Intelligence (TPAMI)*, 22(11):1330–1334.

Zhang, Z., Rebecq, H., Forster, C., and Scaramuzza, D. (2016). Benefit of Large Field-of-View Cameras for Visual Odometry. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, pages 801–808.

Zhou, T., Brown, M., Snavely, N., and Lowe, D. (2017). Unsupervised Learning of Depth and Ego-Motion from Video. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 6612–6619.

In der Schriftenreihe des Instituts für Geodäsie und Geoinformation der Rheinischen Friedrich-Wilhelms-Universität Bonn sind erschienen:

2699-6685 (Online)

Heft 67 Johannes Schneider
2020 Visual Odometry and Sparse Scene Reconstruction for UAVs
with a Multi-Fisheye Camera System

Heft 66 Emanuele Palazzolo
2020 Active 3D Reconstruction for Mobile Robots

Heft 65 Olga Vysotska
2020 Visual Place Recognitionin Changing Environments

Heft 64 Kaihong Huang
2020 Extrinsic Calibration and Ego-Motion Estimation for Mobile Multi-Sensor Systems

Heft 63 Christian Merfels
2020 Sensor fusion for localization of automated vehicles

Heft 62 Igor Bogoslavskyi
2020 Robot Mapping and Navigation in Real-World Environments

Heft 61 Johann Christian Rose
2020 Automatische, hochaufgelöste 3D-Phänotypisierung von
Trauben und Beeren der Weinrebe unter Feldbedingungen

Heft 60 Florian Zimmermann
2020 Analysis and mitigation of site-dependent effects in static and kinematic
GNSS applications

ISSN 1864-1113

Heft 59 Sebastian Halsig
2018 Atmospheric refraction and turbulence in VLBI data analysis

Heft 58 Sebastian Alexander Kropp
2017 Hochwasserrisiko und Immobilienwerte

Heft 57 Sujit Kumar Sikder
2017 Exploring Urban Structure to Approach Energy Optimization

Heft 56 Asad Asadzadeh
2017 Conceptualizing the concept of disaster resilience:
a hybrid approach in the context of earthquake hazard

Heft 55 Stefan Paulus
2017 Potentiale von Laserscannern zur Phänotypisierung von Pflanzen für
den Einsatz im Hochdurchsatz-Screening

Heft 54 Proceedings of the First International Workshop on VLBI Observations of
2017 Near-field Targets

Heft 53 Christian Eling
2016 Entwicklung einer direkten Georeferenzierungseinheit zur Positions-
und Orientierungsbestimmung leichter UAVs in Echtzeit

Heft 52 Maike Schumacher
2016 Methods for assimilating remotely-sensed water storage changes into
hydrological models

Heft 51 Christoph Holst
2015 Analyse der Konfiguration bei der Approximation ungleichmäßig abgetasteter Oberflächen
auf Basis von Nivellements und terrestrischen Laserscans

Heft 50 2015	Lutz Rolf Roese-Koerner Convex Optimization for Inequality Constrained Adjustment Problems
Heft 49 2015	Jan Martin Brockmann On High Performance Computing in Geodesy Applications in Global Gravity Field Determination
Heft 48 2015	Judith Leek The application of impact factors to scheduling VLBI Intensive sessions with twin telescopes
Heft 47 2015	Thomas Artz Determination of Sub-daily Earth Rotation Parameters from VLBI Observations
Heft 46 2015	Roelof Rietbroek Retrieval of Sea Level and Surface Loading Variations from Geodetic Observations and Model Simulations: an Integrated Approach
Heft 45 2014	Ehsan Forootan Statistical Signal Decomposition Techniques for Analyzing Time-Variable Satellite Gravimetry Data
Heft 44 2014	Erich Weiß Lebensbilder der preußischen Verwaltung des 19. und 20. Jahrhunderts im Wandel Eine Sammlung biographischer Miniaturen
Heft 43 2014	Neysa Jacqueline Setiadi Assessing People´s Early Warning Response Capability to Inform Urban Planning Interventions to Reduce Vulnerability to Tsunamis Case Study of Padang City, Indonesia
Heft 42 2013	Nils Leber Entwicklungsperspektiven metropolitaner Peripherien im Rahmen Stadtregionaler Planungs- und Entwicklungsprozesse am Beispiel Nordrhein-Westfalen
Heft 41 2013	Sophie Schetke Socio-environmental impacts of settlement growth under conditions of fostered infill development: a methodological framework for a multicriteria assessment
Heft 40 2013	Ribana Roscher Sequential Learning Using Incremental Import Vector Machines for Semantic Segmentation
Heft 39 2013	Michael Ying Yang Hierarchical and Spatial Structures for Interpreting Images of Man-made Scenes Using Graphical Models
Heft 38 2013	Sabine Daniela Bauer Automatische Detektion von Krankheiten auf Blättern von Nutzpflanzen
Heft 37 2013	Martin Drauschke Ein hierarchischer Ansatz zur Interpretation von Gebäudeaufnahmen
Heft 36 2013	Timo Dickscheid Robust Wide-Baseline Stereo Matching for Sparsely Textured Scenes
Heft 35 2013	Alexander Barth Vehicle Tracking and Motion Estimation Based on Stereo Vision Sequences
Heft 34 2013	Richard Steffen Visual SLAM from image sequences acquired by unmanned aerial vehicles
Heft 33 2013	Till Rumpf Finding spectral features for the early identification of biotic stress in plants

Heft 32 2012	Christian Siemes Digital Filtering Algorithms for Decorrelation within Large Least Squares Problems
Heft 31 2012	Silvia Becker Konsistente Kombination von Schwerefeld, Altimetrie und hydrographischen Daten zur Modellierung der dynamischen Ozeantopographie
Heft 30 2013	Annette Eicker / Jürgen Kusche (eds.) Lecture Notes from the Summer School of DFG SPP1257 Global Water Cycle
Heft 29 2012	Matthias Siemes Ein Beitrag zur koordinatengesteuerten Aussaat von Rübenpflanzen mittels Multi-Sensor-System und Filteransatz
Heft 28 2012	Jörg Schmittwilken Attributierte Grammatiken zur Rekonstruktion und Interpretation von Fassaden
Heft 27 2012	Markus Rembold Die Anerkennung und Feststellung von Grundstücksgrenzen Ein Beitrag zur Entwicklung des Liegenschaftskatasters im Lande Nordrhein-Westfalen in Vergangenheit, Gegenwart und Zukunft
Heft 26 2012	Lihua Li Separability of deformations and measurement noises of GPS time series with modified Kalman filter for landslide monitoring in real-time
Heft 25 2012	Benedikt Frielinghaus Ökonomisches Entscheidungstool zur Wohnbaulandentwicklung Wirtschaftlichkeitsanalysen potenzieller Wohnbauflächen auf der Ebene des Flächennutzungsplanes
Heft 24 2011	Enrico Kurtenbach Entwicklung eines Kalman-Filters zur Bestimmung kurzzeitiger Variationen des Erdschwerefeldes aus Daten der Satellitenmission GRACE
Heft 23 2011	Sarah Böckmann Robust determination of station positions and Earth orientation parameters by VLBI intra-technique combination
Heft 22 2011	20th Meeting of the European VLBI Group for Geodesy and Astronomy Proceedings
Heft 21 2011	Philipp Zeimetz Zur Entwicklung und Bewertung der absoluten GNSS-Antennenkalibrierung im HF-Labor
Heft 20 2011	Alessandra Roy Effects on the Geodetic-VLBI Observables Due to Polarization Leakage in the Receivers
Heft 19 2011	Dietmar Weigt Auswirkungen von Flughäfen insbesondere von Fluglärm auf den Immobilienmarkt am Beispiel des Marktsegments „individuelles Wohnen“
Heft 18 2011	Anno Löcher Möglichkeiten der Nutzung kinematischer Satellitenbahnen zur Bestimmung des Gravitationsfeldes der Erde
Heft 17 2010	Basem Elsaka Simulated Satellite Formation Flights for Detecting the Temporal Variations of the Earth’s Gravity Field
Heft 16 2010	2nd International Conference on Machine Control & Guidance Proceedings
Heft 15 2009	Alexandra Weitkamp Brachflächenrevitalisierung im Rahmen der Flächenkreislaufwirtschaft

Heft 14 2008	Akbar Shabanloui A New Approach for a Kinematic-Dynamic Determination of Low Satellite Orbits Based on GNSS Observations
Heft 13 2008	Frank Friesecke Stadtumbau im Konsens!? Zur Leistungsfähigkeit und Fortentwicklung des städtebaulichen Instrumentariums unter Schrumpfungsbedingungen
Heft 12 2008	Heinz Rütz Zur Kostenanalyse der privaten Umlegung als Teil der konsensualen integrierten Baulandentwicklung
Heft 11 2008	Gaby Alexandra Boele-Keimer Kommunales Kennzahlenmanagement am Beispiel von Vermessungs- und Katasterämtern in Nordrhein-Westfalen
Heft 10 2008	Annette Eicker Gravity Field Refinement by Radial Basis Functions
Heft 9 2008	Torsten Mayer-Gürr Gravitationsfeldbestimmung aus der Analyse kurzer Bahnbögen
Heft 8 2008	Boris Kargoll On the Theory and Application of Model Misspecification Tests
Heft 7 2008	Hamza Alkhatib On Monte Carlo Methods
Heft 6 2008	Klaus Borchard Annäherungen an Städtebau und Raumentwicklung
Heft 5 2008	Jens Jähnke Zur Teilmarktbildung beim Landerwerb der öffentlichen Hand
Heft 4 2008	Atef Abd-Elhakee Makhloof The Use of Topographic Isostatic Mass Information
Heft 3 2008	Markus Vennebusch Singular Value Decomposition and Cluster Analysis
Heft 2 2007	Christian Beder Grouping Uncertain Oriented Projective Geometric Entities
Heft 1 2007	Klaus Börger Geodäsie und Quantenphysik

2699-6685 (Online)