# A Specialized Kalman Filter Framework for IMU Aided Stereo SLAM

Dipl.-Inf. Arne Petersen

Dissertation
zur Erlangung des akademischen Grades
Doktor der Ingenieurwissenschaften
(Dr.-Ing.)
der Technischen Fakultät
der Christian-Albrechts-Universität zu Kiel
eingereicht im Jahr 2013

Kiel Computer Science Series (KCSS) 2014/3 v1 dated 2014-03-30

ISSN 2193-6781 (print version)
ISSN 2194-6639 (electronic version)

Electronic version, updates, errata available via https://www.informatik.uni-kiel.de/kcss

The author can be contacted via mip.informatik.uni-kiel.de

Published by the Department of Computer Science, Kiel University

Multimedia Information Processing - MIP

Please cite as:

▷ Arne Petersen. *A Specialized Kalman Filter Framework for IMU Aided Stereo SLAM* Number 2014/3 in Kiel Computer Science Series. Department of Computer Science, 2011. Dissertation, Faculty of Engineering, Kiel University.

```
@book{Petersen:KCSS:2014:3,
  author    = {Petersen, Arne},
  title     = {A Specialized Kalman Filter Framework for IMU Aided Stereo SLAM},
  publisher = {Department of Computer Science, CAU Kiel},
  year      = {2014},
  number    = {2014/3},
  series    = {Kiel Computer Science Series},
  note      = {Dissertation, Faculty of Engineering,
               Kiel University.}
}
```

1. Gutachter: Prof. Dr.-Ing. Reinhard Koch
Christian-Albrechts-Universität
Kiel

2. Gutachter: Prof. Dr. Didier Stricker
Deutsches Forschungszentrum für Künstliche Intelligenz
Kaiserslautern

3. Gutachter: **NA**

Datum der mündlichen Prüfung: 1. Oktober 2013

"... Wenn Ihr den Rundfunk höret, so denkt auch daran, wie die Menschen in den Besitz dieses wunderbaren Werkzeuges der Mitteilung gekommen sind. Der Urquell aller technischen Errungenschaften ist die göttliche Neugier und der Spieltrieb des bastelnden und grübelnden Forschers und nicht minder die konstruktive Phantasie des technischen Erfinders. ..."

Albert Einstein
7. Deutsche Funkausstellung und Phonoschau
Berlin, 22. August 1930

# Zusammenfassung

Diese Arbeit hat die Entwicklung, Implementierung und Analyse eines Verfahrens zur simultanen Lokalisierung und Kartierung, englisch kurz SLAM, zum Ziel. Dazu wird ein Hard- und Software-System entwickelt und analysiert, welches mittels zweier Kameras (stereo System) und eines Inertialsensors (IMU) die Umgebung beziehungsweise die Eigenbewegung wahrnimmt. Diese Informationen werden dazu verwendet, Landmarken zu erstellen, welche eine Karte der Umgebung bilden. Simultan dazu wird die Position und Ausrichtung relativ zu dieser Karte bestimmt.

Der Kern der Innovation dieser Arbeit ist die Entwicklung einer neuen Parametrisierung für Kalman Filter basierte SLAM Algorithmen. Sie ist spezialisiert für den Einsatz mit stereo Kamerasystemen und besonders geeignet für die Verwendung von Inertialsensorik. Zusätzlich wird die Kalibrierung der Kameras zur angebundenen IMU in das SLAM Verfahren integriert. Zur Validierung werden die verwendeten Modelle intensiv analysiert und mit "state of the art" Methoden verglichen. Desweiteren wird eine detailierte Beschreibung der verwendeten Modelle und der Software Implementierung gegeben.

Die Arbeit gibt im Kapitel 1 eine Einführung in die Aufgabenstellung SLAM und eine Zusammenfassung der vorhergehenden Forschung in diesem Bereich. Des Weiteren wird ein Überblick über die Struktur der Arbeit gegeben und die wesentlichen wissenschaftlichen Neuerungen zusammengefaßt. Kapitel 2 führt die verwendeten Notationen ein und stellt die benötigten Grundlagen zur Verfügung. Anschließend gibt Kapitel 3 eine Einführung in die Theorie der linearen Schätzer. In 4 werden schließlich das Hardwaresystem und die theoretischen Modelle, welche die Grundlage des SLAM-Verfahrens bilden, eingeführt. Die vorgestellten Modelle werden anschließend in Kapitel 5 analysiert und verglichen. In Kapitel 6 wird das erstellte Software System detailiert beschrieben und in 7 intensive synthetische und reale Tests des vollen Systems durchgeführt. Den Abschluß in Kapitel 8 bildet ein Resumee und ein Ausblick auf zukünftige Forschungsmöglichkeiten.

# Abstract

This thesis aims at the design, implementation and analyzation of a method for **S**imultaneous **L**ocalization **A**nd **M**apping. For that purpose a hard- and software system is developed and analyzed, that makes use of a pair of stereo cameras and an inertial measurement unit. This enables the system to perceive it's environment and ego-motion respectively. This information is used to build up a sparse environment map (**M**apping) and estimate the traveled trajectory (**L**ocalization). Moreover, the relative pose between the camera and the IMU (camera to IMU calibration) is estimated. The main focus is, to use a novel state parameterization improving the estimations consistency. That is, the appraisal of estimation errors is more reliable compared to other representations. At first the task of SLAM, previous work and the thesis' structure are discussed. Following that chapter 2 describes the used notation and theoretic fundamentals and chapter 3 the theory of linear probabilistic estimators. Chapter 4 introduces the developed framework and it's models, which are analyzed in part 5. After a detailed description of the systems implementation an experimental evaluation is performed in chapter 7. The thesis closes with a conclusion and an outlook to future work.

# Danksagung

Zuerst möchte ich meinem Doktorvater Prof. Dr.-Ing. Reinhard Koch für seine Unterstützung und das entgegengebrachte Vertrauen danken. Ich werde mich immer an die überaus lehrreiche Zeit und die freundschaftliche Atmosphäre in unserer Arbeitsgruppe erinnern. In diesem Sinne bin ich auch allen Kollegen für die an- und auf-regenden Diskusionen dankbar. Diese haben mich nicht selten zu neuen Ideen inspiriert. Einen besonderen Dank auch an Torge Storm und Renate Staecker für die wiederholte technische Notfallhilfe und den Beistand bei den bürokratischen Herausforderungen.

Erwähnen möchte ich an dieser Stelle auch die gute Zusammenarbeit mit Dr.-Ing. Marc-André Beyer während unseres gemeinsamen Forschungsprojekts. Dank dieser konnte ich unter der schwierigen Aufgabenstellung den Grundstein zu dieser Arbeit finden. Außerdem möchte ich Frau Maren Loose für ihre Geduld bei der sprachlichen Korrektur dieser Schrift danken.

Zum Schluß, aber nicht letztens, danke ich meiner Familie. Ihr alle habt mich auf meinem Weg begleitet und standet mir zur Seite, wann immer es nötig aber nicht unbedingt einfach war. Besonders gilt dies für meine Eltern. Eure Unterstützung und Motivation haben mir die Freiheit gegeben, welche diese Arbeit überhaupt erst möglich gemacht hat. Abschließend ein Extra an meinen Schwager Oke Jens für die graphische Aufwertung des Einbandes.

# Contents

# List of Figures

# List of Tables

# Abbreviations

| | |
|---|---|
| ATE | **A**bsolute **T**rajectory **E**rror |
| DOF | **D**egrees **O**f **F**reedom |
| EKF | **E**xtended **K**alman **F**ilter |
| ES | **E**clidean **S**pace landmarks |
| GPU | **G**raphics **P**rocessing **U**nit |
| ID | **I**verse **D**epth landmarks |
| IEKF | **I**terated **E**xtended **K**alman **F**ilter |
| IMU | **I**nertial **M**easurement **U**nit |
| KDE | **K**ernel **D**ensisty **E**stimation |
| KF | **K**alman **F**ilter |
| KLT | **K**anade-**L**ucas-**T**omasi |
| NEES | **N**ormalized **E**stimation **E**rror **S**quared |
| MEMS | **M**icro **E**lectro-**M**echanical **S**ystem |
| PD | **P**oint **D**isparity landmarks |
| PDF | **P**robability **D**ensity **F**unction |
| RMSE | **R**oot **M**ean **S**quare **E**rror |
| RP | **R**andom **P**rocess |
| RPE | **R**elative **P**ose **E**rror |
| SIFT | **S**cale **I**variant **F**eature **T**ransform |
| SLAM | **S**imultaneous **L**ocalization **A**nd **M**apping |
| TFT | **T**ri**F**ocal **T**ensor |
| WLSE | **W**eighted **L**east **S**quares **E**stimation |
| $[b]$ | units of distances in baselines of stereo rig |
| $[npx]$ | units of normalized pixel coordinates |
| $[\cdot]$ | units of an SI (metric) entity |

# Symbols

| | |
|---|---|
| $\in$ | "element in set" relation |
| $\forall$ | "for all" in symbolic logic |
| $\exists$ | "exists" in symbolic logic |
| $\wedge$, $\vee$ | "and", "or" relations in symbolic logic |
| $\mathtt{n}, \mathtt{N}$ | cardinal number |
| $\mathbb{N}$ $(\mathbb{N}_0)$ | the set of integers $> 0$ $(\geqslant 0)$ |
| $\lvert M \rvert$ | cardinal numeral of finite set $M$ |
| $2^M$ | power set: set of all sub sets of $M$ |
| $\mathbb{R}$ $(\mathbb{R}^+)$ | field of real numbers $(\geqslant 0)$ |
| $[a, b]$, $(a, b)$ | set real numbers $r$ with $a \leqslant r \leqslant b$, $a < r < b$ |
| $f(\cdot)$ | template parameter for function model $f$ |
| $f \circ g$ | concatenation of functions $f$ and $g$, that is: $(f \circ g)(\cdot) = f(g(\cdot))$ |
| $\mathcal{B}_{(f)}$ | range of function $f$, i. e., set of all images of $f$ |
| $\mathbb{R}^{\mathtt{n}}$ | n-dimensional Euclidean space (n-space) |
| $[\cdot]_{\equiv}$ | equivalence class modulo the relation $\equiv$ |
| | |
| $0_{\mathtt{n}}$ | n-dimensional zero vector |
| $\mathtt{e}_{\mathtt{i}}^{\mathtt{n}}$ | $\mathtt{i}^{\text{th}}$ vector of $\mathbb{R}^{\mathtt{n}}$ orthonormal basis |
| $\mathtt{e}_1, \mathtt{e}_2, \mathtt{e}_3$ | orthonormal basis for $\mathbb{R}^3$ |
| $x, y$ | points in 2-space |
| $X$ | point in 3-space |
| $v_{(\mathtt{i})}, v_{(\tau)}, v_{(\mathtt{i},\tau)}$ | for a vector $v$: component $\mathtt{i}$, value at time $\tau$ and component $\mathtt{i}$ at time $\tau$ |
| $\lvert\cdot\rvert_{\mathtt{n}}$ | the $\mathtt{n}$-norm of a vector or matrix |
| $\otimes$ | vector cross product in $\mathbb{R}^2$ and $\mathbb{R}^3$ respectively |
| $\odot$ | vector outer product |

| | |
|---|---|
| $\mathbb{R}^{\mathtt{n}\times\mathtt{m}}$ | space of linear mappings $\mathbb{R}^n \rightarrow \mathbb{R}^m$ |
| $0^{\mathtt{n}}$ | $\mathtt{n} \times \mathtt{n}$ zero matrix |
| $1^{\mathtt{n}}$ | $\mathtt{n} \times \mathtt{n}$ identity matrix |
| $P_{12}$ | projection matrix $\left(\begin{smallmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{smallmatrix}\right)$ |
| $P_{\mathtt{i}}$ | projection matrix $\mathtt{e}_{\mathtt{i}}^{3T}$ |
| $\lvert A \rvert$ | determinant of matrix $A$ |
| $diag\,(A)$ | vector of diagonal entries of matrix $A$ |
| $eig\,(A)$ | vector of eigenvalues of $A$ (sorted ascending) |
| | |
| $\mathbb{P}^{\mathtt{n}}$ | n-dimensional projective space |
| $\mathbb{P}^2_\star$ | space dual to $\mathbb{P}^2$ - set of lines in a specific plane |
| $\mathbf{x}$ | point in $\mathbb{P}^2$ |
| $\mathbf{X}$ | point in $\mathbb{P}^3$ |
| $\mathbb{P}^{\mathtt{n}\times\mathtt{m}}$ | projective equivalence classes of $\mathbb{R}^{(\mathtt{n}+1)\times(\mathtt{m}+1)}$ |
| $\mathbf{A}$ | arbitrary element of $\mathbb{P}^{\mathtt{n}\times\mathtt{m}}$ (all but $\mathbf{X}$) |
| $\mathbf{P}$ | homogeneous projection matrix |
| $\mathbf{H}\,(\cdot)$ | homogenization of projective point |
| $\mathcal{T}$ | TriFocal Tensor |
| | |
| $\hat{X}$ | random variable |
| $\bar{X}$ | mean of the random variable $\hat{X}$ |
| $X$ | sample drawn from the random variable $\hat{X}$ |
| $\varphi_X$ | PDF of a random variable |
| $\hat{\varphi}_X$ | empirical PDF of a random variable |
| $C_X$ | covariance matrix for random variable $X$ |
| $\hat{C}_X$ | empirical covariance matrix for random variable $X$ |
| $C_{X,Y}$ | correlation matrix for random variables $X$ and $Y$ |
| $\mathcal{N}\left(\bar{X}, C_X\right)$ | Gaußian (normal) distribution, mean $\bar{X}$, covariance $C_X$ |
| $\hat{p}, \bar{p}, p$ | estimation parameters as RV, ground truth and estimation |
| | |
| $\mathcal{I}\,(M, C)$ | informativity of a set $M$ assuming covariance $C$ |
| $\mathcal{LE}_f$ | linearization error for a function $f$ |

| | |
|---|---|
| $\mathcal{X}$ | landmark vectors |
| $\mathcal{L}$ | landmark descriptors |
| | |
| $R_a^b$ | matrix rotating a point in coordinates of $a$ to $b$ |
| $R_{(\phi(\mathtt{i}))}$ | matrix for rotation of angle $\phi(\mathtt{i})$ along $\mathtt{e_i}$ |
| $R_{(\phi)}$ | $= R_{(\phi(3))} \cdot R_{(\phi(2))} \cdot R_{(\phi(1))}$ rotation induced by $\phi$ |
| $\Phi_a^b$ | incremental rotation corresponding to $R_a^b$ |
| $t_{ab}^c$ | translation from $a$ to $b$ in coordinates of $c$ |
| $v_{ab}^c$ | velocity of $b$ with respect to $a$ in coordinates of $c$ |
| $a_{ab}^c$ | acceleration of $b$ with respect to $a$ in frame $c$ |
| $w_{ab}^c$ | turn rates of $b$ with respect to $a$ in frame $c$ |
| $\Gamma_a^b = \left\langle \Phi_a^b, t_{ab}^a \right\rangle$ | transformation of coordinates from $a$ to $b$ |
| $\tau$ | continuous time |
| $\tau_{\mathtt{k}}$ | discrete time index, $\mathtt{k}$ when clear from the context |

Chapter 1

# Introduction

## 1.1 Motivation

During the past decades the need of automated navigation for unmanned vehicles received increasing attention. Such independent platforms are used, to solve various tasks in diverse environments. These range from robots aiding industrial manufacturing in known environments, to multi-sensor platforms for exploration of deep sea or extraterrestrial areas. For the latter it is necessary, that the system is robust and adaptive. Beside the need of special hardware a software has to be provided, which is fault tolerant and able to adapt to new environment conditions. The basis for automated navigation is the determination of the systems position and orientation, pose for short, and a map of its environment. By combining the platforms localization and the environment mapping the **S**imultaneous **L**ocalization **A**nd **M**apping, SLAM for short, arises.

To enable the navigating systems to observe their environment, sensors are needed. Sensors are all kinds of devices, that deliver information on the systems conditions and the "outer world". On the one hand, cameras or similar sensors are able, to gather information on the environment, that is to be mapped. Such sensors observe the surrounding area directly, but they provide only indirect information on the vehicle's state. Other sensors such as GPS or altimeters observe the vehicles state but not its environment. The combination of diverse sensors benefits the localization and mapping by exploiting the advantages and overcome the drawbacks of single devices. Fusing the data delivered by the different sensors is called sensor fusion. It is able to provide information from all involved devices, being more than just the sum of all gathered data. Thanks to the fact, that the costs and size for all these sensing devices dramatically decreased in the past decades,

multiple sensors can be used for a single SLAM platform. By this powerful tools arise, that allow for creation of detailed environment maps and precise system localization.

Stereo cameras have proven, to be valuable tools for pose estimation and map building. They gather information on the environment not only by a single 2D image but are also able to percept depth directly. Even though other sensors, such as laser scanner, radar and the like, provide information more directly (without additional information processing), they are far more expensive. On the one hand, vision systems, i. e., stereo systems, are able to precisely determine slow movements over a long time. On the other hand, low cost inertial measurement units provide information on fast movements for a short time. Thus, the combination of both sensor types is self-evident for low cost SLAM systems. Therefore, the design of an IMU aided stereo SLAM system is the first major objective in this thesis.

Beside estimating the systems poses and environment maps predicting error bounds is a major challenge. They provide valuations to the magnitude of estimation errors. Since SLAM is supposed to be integrated in an automated navigation process, such information are of high value. For example the decision, whether the system is able to pass a mapped door, not only depends on its estimated pose and map but also on the uncertainty of the estimation. Being sure, the position is accurate to a few centimeters, the door can be passed without danger. Having an uncertainty of a few meters the door can hardly be classified, to be an/no obstacle. Analyzing the quality of such error valuations for IMU aided stereo SLAM systems, is this thesis second major objective.

## 1.2 Previous Work

Various kinds of estimators have been designed, to solve the task of SLAM. These range from Kalman Filters (see [Sch+07], [SSM06]), covariance intersection methods [JU01] and particle filters [İB09], to non linear estimation of quadrifocal tensors [HK10]. Nevertheless the most commonly used estimators are based on the famous **K**alman **F**ilter (see [Kal60], [WB06]), KF for short. This is due to the fact, that SLAM benefits from the KFs real-time capability and the suitability of KFs to sensor fusion. The ability, to not only

estimate the system state but also to provide information on the estimations reliability, is another reason for its popularity.

Since a major task of SLAM is sensor fusion for navigation in unknown environment, systems for different environments and with various sensors have been developed. The application areas range from wheel driven indoor and outdoor navigation to extraterrestrial exploration. In the 1980s Brooks (see [Bro86], [Bro85]) developed a generalized control system for mobile robots, and proposed a system for indoor map building using sonar scanner and stereo cameras. In [Dis+01] Dissanayake et al. propose a car mounted SLAM system fusing environment information from a radar scanner and additional information on the cars' movement. In [Hyg+04] and [SF08] airborne SLAM systems are discussed, to visually create a 3D map of surrounding area. Even for underwater application SLAM solutions were discussed, as is done by Hildebrandt et al. in [HK10]. The research discussed in these publications proved the applicability of the SLAM idea to such distinct problem statements.

Apart from the used estimators and sensors the solutions to the SLAM problem differ in the representation of landmarks. Landmarks are distinctive entities in the environment, i. e., 3-space points for visual SLAM, that can be tracked, using the equipped sensors. For visual SLAM the so called **I**nverse **D**epth (**ID**) representation has become a state of the art parametrization, e. g., see [MCD06], [CDM08], [SLP07]. For this model, a landmark is represented by the position of the camera, from where the point was observed first, the direction to the 3-space point and its inverse Euclidean distance to the camera center. A drawback of this parametrization is, that it uses a 6 dimensional representation. Since a point in 3D-space only has 3 degrees of freedom (DOF), it is over parametrized. Thus, ambiguities in the estimation can arise. In especially points initialized from the same pose can end up with different positions for the respective camera. Montiel et al. (see [CDM08]) have shown, that in ID the error propagation for depth is nearly linear under certain assumptions. Nevertheless, the models include inverse tangent and normalization functions for landmark initialization resulting in non linearities. To the authors knowledge, these have not yet been discussed by Montiel et al. or others.

In [Paz+08] Paz et al. discuss deficiencies of the ID parametrization, when estimating landmarks for short camera distances. For their stereo

SLAM system they partition landmarks in far away and nearby landmarks. For far points ID is used, and the 3 dimensional **E**uclidean **S**pace (**ES**) for nearby points. This improves the estimation of landmarks and the system pose. Nevertheless, as can be seen in their evaluation, the parametrization still has problems covering the true error distribution for reconstructed 3-space points. Another benefit of combining ES and ID is the reduction of computational effort, as discussed by Paz et al. and Montiel et al. Since the three parameters for ES induce a reduced memory consumption and runtime, compared to ID, Montiel et al. propose in [CDM08] a "switch" from ID to ES. They deduced a linearity index, allowing for determining conditions for an appropriate representation of 3-space points by ES. This information is used, to convert a landmark from ID to ES as soon as applicable.

**Stereo SLAM** Due to the additional information provided by stereo camera systems, various stereo SLAM systems have been developed, e. g., see [Paz+08], [Sch+07]. Such systems have in common, that the cameras are used independently. That is, the SLAM is improved, compared to monocular approaches, by using two observations for each landmark. Even though the known transformation between the stereo cameras is utilized, certain vision constraints are disregarded. At this point Solà et al. (see [SMD07]) claim, that using two independent cameras outperforms stereo rigs. In fact, they propose a combination of monocular and stereo vision, resulting in improved estimation. Since this is a combination of both models, this approach can be used to fuse most monocular and stereo models.

Indeed more sophisticated approaches were made, explicitly exploiting the constraints of stereo cameras. As mentioned above, Hildebrandt et al. discussed an underwater SLAM system in [HK10]. They make use of the stereo system, by estimating a quadrifocal tensor, constraint by the known stereo calibration. In [HKD06] a representation, similar to the one proposed in this thesis, has been proposed for observation models. In contrast to our work, they do not use this for modeling landmarks in the SLAM system, but only for using stereo constraints in the observation models. It was shown, that such observation representations fulfill the Gaußian noise assumption and help to improve the systems performance.

**IMU aided SLAM** As for stereo vision, the inertial measurement units have a long history in SLAM. The basic difference between the approaches is the type of IMU integration. In [PWA10] Nourani-Vatani et al. propose a combination of an IMU, a car steering model and vision information for estimation of car trajectories. To aid their navigation system, they use the horizontal orientation, delivered by the IMU. Other systems, such as [HK10], use a motion prediction by IMUs to provide initial poses for non linear estimators.

The most sophisticated method for integrating IMU measurements to navigation tasks is strap down navigation. It makes use of acceleration and turn rate measurements, provided by the IMU, to determine the position and orientation based on the preceeding pose. Such methods are, e. g., used in [JS11] and [CP04]. Since this is well suited to the KFs prediction-update procedure, it is predestined for such filters. In [Ble09] Bleser discussed different approaches of IMU integration. The conclusion was drawn, that strap down navigation like fusion of vision and inertial information, performs best among the studied methods.

## 1.3 Contribution

This thesis aims at the design and analysis of IMU aided stereo SLAM systems, including IMU to camera auto calibration. It provides four major contributions. These are a new landmark parametrization, a detailed model design derivation, the implementation in a filter framework and an extensive analysis of estimation precision and consistency.

Therefore, the proposed landmark parametrization is compared to state of the art methods. This is done by comparing the models analytically and in synthetical tests. To allow for the IMU to camera auto-calibration, the proposed and state of the art methods are augmented by the respective calibration models. Moreover, real world experiments are carried out, to prove the systems applicability to modern SLAM tasks.

As will be shown, the proposed system is able to stably estimate the full navigation state and calibration with high precision. It stays consistent and reliable even for long term navigation. Its robustness and real-time applicability allows for applying the system to free hand movements under

difficult visual conditions.

Parts of this work have been published in [PK12a], [PK12b], [PB11] and [PK10].

## 1.4 Outline

The thesis is structured as follows. Chapter 2 introduces the fundamentals, needed for the following derivations and discussions. These are especially the basic conventions on notation and coordinate systems, as a central part of model derivations. Moreover, elementary models for sensor devices and information processing are discussed.

In chapter 3, the estimation theory, used in the proposed filters, is discussed. After a short introduction to probability theory and linear probabilistic estimation, the Kalman Filter theory is discussed. Following that, some tools for the analysis of the discussed parametrization models are introduced.

Chapter 4 explains the decision of hardware design, this thesis models are embedded in. Moreover, it defines the newly developed state parametrization model PD, and provides an overview on two state of the art methods. Finally, the error models, used for the proposed Kalman Filter, are introduced.

The discussion on system modeling is followed by an analysis of their properties and applicability to linear estimation in chapter 5. Moreover, the advantages and drawbacks of the different parameterizations are discussed.

In chapter 6 the embedding of the proposed SLAM filter in a software framework is discussed. A detailed description is given, that clarifies the algorithms work flow and internal structure. This includes the data structures, a semantic module structure and a detailed work flow of implemented models. Additionally, a camera to IMU calibration software is introduced.

In chapter 7 the proposed SLAM systems are validated experimentally. This is done using synthetical as well as real world datasets. The synthetical tests are restricted to the landmark and pose models, i. e., no IMU modeling is done. The first set of real world experiments is carried out on an offline dataset, providing ground truth information. Thus, the IMU models and their combination with the vision models can be evaluated. In the second set the SLAM systems are applied to a free hand movement scenario, validating

their real-time capabilities and its robustness in uncontrolled environments.

The thesis closes with a conclusion in chapter 8, providing a summary of the discussed work and an outlook to future work.

Chapter 2

# Fundamentals

This chapter provides the fundamentals and the according notations used throughout this thesis. The projective geometry is used to describe the theory of imaging using the well known pinhole camera and is introduced first. Afterwards an introduction to the probability theory is given, which is needed for the basics of the used estimation methods. Since the pinhole model is an ideal mathematical model, the additional knowledge needed for real live camera systems is described in the succeeding section. Finally the principles of inertial navigation are introduced.

## 2.1 Basic Notations

Conventions on mathematical notations are used throughout this thesis, to ease the readability of complex or repeatedly used formulae. These are introduced in the following.

To ease the readability of formulae within a higher level context, column vectors are often written as row vectors. E. g., for vectors $a$, $b$ and $c$

$$\begin{pmatrix} a & b & c \end{pmatrix} \qquad \text{is used to shorten} \qquad \begin{pmatrix} a \\ b \\ c \end{pmatrix}$$

This is especially applied to equations in continuous text, or to those that compose vectors from several entities, since a column vector would exceed the page format. For all involved formulae the original vector size and orientation is clear from the context.
When dealing with matrices, a unified representation is used in the following. In especially a Matrix $A$ is made up of elements $A_{(\mathtt{i},\mathtt{j})} = a^{\mathtt{i}}_{\mathtt{j}}$, where the indices

$\mathtt{i}$ and $\mathtt{j}$ represent the location in the matrices rows and columns respectively. Since block matrices will be used frequently, a shortened notation for accessing sub-matrices is given. Therefore, $A\,(\mathtt{i}:\mathtt{j},\mathtt{k}:\mathtt{l})$ denotes a sub-matrix of $A$ with the top left index $(\mathtt{i},\mathtt{k})$ and the bottom right index $(\mathtt{j},\mathtt{l})$. That is:

$$A_{(\mathtt{i}:\mathtt{j},\mathtt{k}:\mathtt{l})} = \begin{pmatrix} a^{\mathtt{i}}_{\mathtt{k}} & \dots & a^{\mathtt{i}}_{\mathtt{l}} \\ \vdots & \ddots & \vdots \\ a^{\mathtt{j}}_{\mathtt{k}} & \dots & a^{\mathtt{j}}_{\mathtt{l}} \end{pmatrix} \tag{2.1.1}$$

An according notation is used for row and column vectors.

For repetitive equations $\left\{\ A_1 \ \middle|\ \dots \ \middle|\ A_{\mathtt{n}}\ \right\}$ is used to abbreviate the usage of matrices $A_1$ to $A_{\mathtt{n}}$ in $\mathtt{n}$ similar formulae. That is:

$$\begin{aligned} B &= \left\{\ A_1 \ \middle|\ \dots \ \middle|\ A_{\mathtt{n}}\ \right\}\cdot a + b \\ &\downarrow \\ B_{(1:\mathtt{m},1)} &= A_1\cdot a + b \\ &\vdots \\ B_{(1:\mathtt{m},\mathtt{n})} &= A_{\mathtt{n}}\cdot a + b \end{aligned}$$

For the evaluations of the discussed SLAM systems an ordering on matrices is needed. Since only positive semi-/definite matrices are used, the ordering is restricted to these. For a symmetric matrix $A \in \mathbb{R}^{n\times n}$ the **positive semi-/definiteness** is defined by:

$$A \text{ is positive definite} \iff \forall x \in \mathbb{R}^n :\ x^T A\, x > 0 \tag{2.1.2}$$

$$A \text{ is positive semidefinite} \iff \forall x \in \mathbb{R}^n :\ x^T A\, x \geqslant 0 \tag{2.1.3}$$

By this a notion of order on the space of positive semidefinite matrices can be defined. That is, for two positive semidefinite matrices $A$ and $B$:

$$A < B \iff B - A \text{ is positive definite} \tag{2.1.4}$$

$$A \leqslant B \iff B - A \text{ is positive semidefinite} \tag{2.1.5}$$

The definitions (2.1.2) and (2.1.3) are equivalent to all eigenvalues of $A$ being (strictly) positive. Thus, the inverse $A^{-1}$ of a positive definite matrix is also positive definite. This allows for the definition of the generalized vector norm:

$$|x|_A := \sqrt{x^T A^{-1} x} \tag{2.1.6}$$

Note that this definition requires $A$ to be strictly positive (i. e., non singular). Applying this norm to matrices being semidefinite, requires the use of the pseudo inverse $A^+$ of $A$ (see [MFW04], [Koc99]).

## 2.2 Coordinate Frames

Coordinate frames, also called coordinate systems, are used throughout this thesis to describe certain entities in a linear vector space. When entities, given in a frame $a$, are to be used in another frame $b$, the relations between these frames have to be taken into account. Beside rotations $R$ and translations $t$ also velocities $v$ ($w$ for angular velocities), accelerations $a$, scales and more complex relations may be involved. In the following, the only relations considered, are translations and rotations as well as their derivatives.

### 2.2.1 Conventions

The notation for rotations, relative entities and their concatenations (equations (2.2.1) to (2.2.5)) is taken from [Wen07]. It improves the readability of equations, that involve multiple coordinate frames and entities. A rotation between the frames $a$ and $b$ is represented using $R_a^b$, rotating an entity given in coordinates of $a$ to those of $b$. The frame an entity $E$ is given in, is defined by an upper index. That is, $E^a$ represents the entity $E$ in coordinates of frame $a$. Entities as translations, velocities etc. are represented relative between objects (coordinate frames, navigating systems etc.). This is indicated by two lower indices. The entity $E_{AB}$ describes a property of $B$ relative to $A$. Using this notation, we have:

$$E_{AB}^b = R_a^b \cdot E_{AB}^a \tag{2.2.1}$$

Let $E^a_{ab}$ and $E^c_{bc}$ relations between the frames $a$, $b$ and $c$ respectively. These relations can be used to determine the according relation between $a$ and $c$. This is done by unifying the coordinate frame and concatenating the single relations by:

$$E^a_{ac} = E^a_{ab} + E^a_{bc} = E^a_{ab} + R^a_c \cdot E^c_{bc} \tag{2.2.2}$$

It is obvious, that such concatenations can be applied to coordinate transformations also. That is, combining the relations $a \to b$ and $b \to c$ results in:

$$R^c_a \quad = \quad R^c_b \cdot R^b_a \tag{2.2.3}$$

Using (2.2.3) the inverse of a rotation is implied by:

$$R^b_a \cdot R^a_b = R^b_b = \mathbf{I} = R^a_a = R^a_b \cdot R^b_a \quad \Longrightarrow \quad R^b_a = (R^a_b)^{-1} \overset{(*)}{=} (R^a_b)^T \tag{2.2.4}$$

Where $(*)$ results from the fact, that rotation matrices are members of the special orthonormal group (see [BS96]).

For some applications it is necessary to not only apply a coordinate transformation via rotation. In addition the datum point has to be changed, i. e., when referring to global positions and the like. Let $t^a_{aX}$ describe the translation between frame $a$ and a point $X$. When this point is to be used in a frame $b$, it has to be given relative to $b$. Thus, $t^b_{bX}$ has to be determined. This can be done by:

$$\begin{aligned} t^b_{bX} &= t^b_{bX} + t^b_{ab} - t^b_{ab} \quad = \quad t^b_{aX} - t^b_{ab} \\ &= R^b_a \cdot t^a_{aX} + t^b_{ba} \end{aligned} \tag{2.2.5}$$

In addition to the general conventions discussed above, specialized notations for incremental rotations and coordinate transformations have to be defined.

**Incremental rotations** are needed to linearize rotation matrices when estimating the respective orientations. Therefore, a notation is proposed, combining a rotation matrix $R$ and an Euler-angle 3-vector $\phi$. It is defined by:

$$\Phi^c_a := \left\{R^c_b; \phi^b_a\right\} = \left\{R^c_b \cdot R(\phi^b_a); 0\right\} \quad \text{represents} \quad R^c_b \cdot R(\phi^b_a) = R^c_a \tag{2.2.6}$$

Here $R(\phi_a^b) = R(\phi_a^b(3))\cdot R(\phi_a^b(2))\cdot R(\phi_a^b(1)) = R_a^b$ is the rotation matrix induced by $\phi_a^b$. Note that the transposition of an incremental rotation changes the interpretation of the Euler angles since it reverses the matrix multiplication order. Thus, in contrast to the rotation matrices, incremental rotations cannot be inverted by switching their indices, i.e.:

$$\left(R_b^c \cdot R(\phi_a^b)\right)^T = R(\phi_a^b)^T \cdot R_c^b \qquad \Longrightarrow \qquad \Phi_a^{c\,T} \neq \Phi_c^a \tag{2.2.7}$$

This becomes obvious, when derivatives of rotations are determined (see section 3.3.4). The incremental rotation notation can be used for concatenations of rotations, by using concatenated rotation matrices:

$$\Phi_c^e \cdot \Phi_a^c = \left\{R_d^e; \phi_c^d\right\} \cdot \left\{R_b^c; \phi_a^b\right\} \quad \text{represents} \quad R_d^e \cdot R(\phi_c^d) \cdot R_b^c \cdot R(\psi_a^b) \tag{2.2.8}$$

For some applications it is helpful to integrate angle increments within incremental rotations. To ease the readability of such equations an additive notation is defined:

$$\Phi_b^d + \psi_a^b \;=\; \left\{R_c^d; \phi_b^c\right\} + \psi_a^b \;=\; \left\{R_c^d \cdot R_{(\phi_b^c)}; \psi_a^b\right\} \;=\; \Phi_a^d \tag{2.2.9}$$

Since this representation of rotations is not unique, a normalization is needed, that can referred to. In the following, the representation $\{R\cdot R(\phi), 0\}$ is said to be the normalized representation of $\Phi = \{R, \phi\}$.

**Coordinate transformations** Using these notations a coordinate transformation $\Gamma_a^b$ (also called pose) is defined by its incremental orientation and the translation of $b$ relative to $a$:

$$\Gamma_a^b = \left\langle \Phi_a^b, t_{ba}^b \right\rangle \tag{2.2.10}$$

Applying this to equation (2.2.5) results in $\Gamma_a^b\,(t_{aX}^a) = t_{bX}^b$. Concatenating coordinate transforms can be done using:

$$\begin{aligned} \left(\Gamma_b^c \circ \Gamma_a^b\right)(t_{aX}^a) &= \Gamma_b^c\left(\Phi_a^b \cdot t_{aX}^a + t_{ba}^b\right) \\ &= \Phi_b^c \cdot \left(\Phi_a^b \cdot t_{aX}^a + t_{ba}^b\right) + t_{cb}^c \\ &= \Phi_b^c \cdot \Phi_a^b \cdot t_{aX}^a + \Phi_b^c \cdot t_{ba}^b + t_{cb}^c \end{aligned} \tag{2.2.11}$$

$$\Longrightarrow \quad \Gamma_a^c \quad = \quad \left\langle \Phi_b^c \cdot \Phi_a^b,\ \Phi_b^c \cdot t_{ba}^b + t_{cb}^c \right\rangle \tag{2.2.12}$$

### 2.2.2 Important coordinate frames

The coordinate frames, that are used throughout this thesis, are the world coordinate system and the navigation, camera and IMU frames. They are used to relate the different navigation entities.

**The earth coordinate system** $\varepsilon$ is orientated conform the earths north, east and down directions. It is located at the initial systems position. The first and third component correspond to the north and east direction respectively. The second component is parallel to the earths gravity $\gamma^\varepsilon$, pointing down from the frames origin. By this the first ($\mathsf{e}_1$-) and third ($\mathsf{e}_3$-) axis span a plane, that is tangential to the earths surface.

**The environment coordinate system** $e$ is used to define the "world" of the systems environment. The systems initial alignment to the down-axis $\mathsf{e}_2$ in $\varepsilon$ can be determined using the IMU, which implicitly measures $\gamma^\varepsilon$. North and east cannot be aligned to the SLAM system precisely. Thus, a certain angle $\phi_\gamma$ for the down-axis rotation parallel to the gravity results. That is:

$$\Gamma_\varepsilon^e = \left\langle R\left(\begin{pmatrix} 0 \\ \phi_\gamma \\ 0 \end{pmatrix}\right), 0_3 \right\rangle \tag{2.2.13}$$

This induces $\gamma^e = \gamma^\varepsilon$. Thus, as for $\varepsilon$ the first and third axis span a plain tangential to the earths surface. Since $\phi_\gamma$ cannot be determined by the used hardware, the environment coordinate system $e$ is used as absolute pose reference. Due to this, $e$ is also said, to be the global coordinate frame.

**The navigation frames** $n$ describe a local reference frame. When local navigation parametrizations are used (e. g., see chapter 4.2.2), their poses are given relative to this frame. By this the navigation frame changes over time, to update the local to global frame reference. When not stated differently, $n$ corresponds to the camera frame of the previous time step. In global parametrizations, $n$ and $e$ are identified with each other, since no local reference is used.

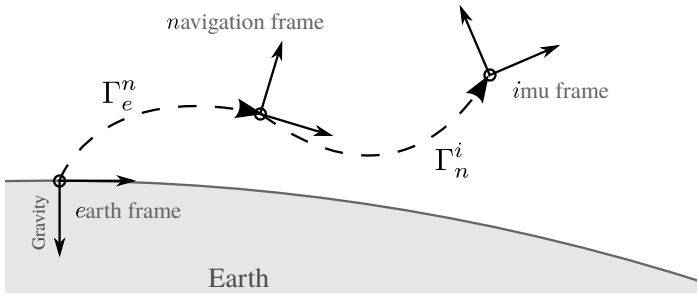


**Figure 2.1.** Visualization of the basic coordinate frames used for navigation.

**The IMUs frame** $i$ is the IMUs body fixed coordinate system. The data provided by the inertial sensor relates to this frame. Moreover, in the proposed navigation parametrizations (see chapter 4.2.5) this frame represents the system pose. The global system pose is given by $\Gamma_e^i = \langle \Phi_e^i, t_{ei}^e \rangle$, the local pose by $\Gamma_n^i = \langle \Phi_n^i, t_{ni}^n \rangle$.

**The camera frames** $c, c'$ are the cameras body fixed coordinate system. The master camera $c$ and the slave camera $c'$ are rigidly coupled, thus, $\Gamma_c^{c'}$ is time invariant. For a detailed description of this coordinate transformation see chapter 2.3.3 and 4.1.1. The visual measurements to be used for localization and mapping are provided in this frame. The transformation $\Gamma_c^i = \langle \Phi_c^i, t_{ci}^c \rangle$ to the $i$-frame represents the IMU-camera calibration. Since the cameras are assumed to be rigidly coupled with the IMU, $\Gamma_c^i$ is time invariant.

The relations between $e$, $n$ and $i$ are visualized in figure 2.1. The relations between $i$ and $c$ (the camera-IMU calibration) are discussed further in sections 4.1.2 and 6.9.

## 2.3 Pinhole Camera

The pinhole camera uses geometrical optics (see [Cha05]) to describe a strongly simplified model for optical imaging. In the following, this projection model and the according mathematical notation, called projective geometry,

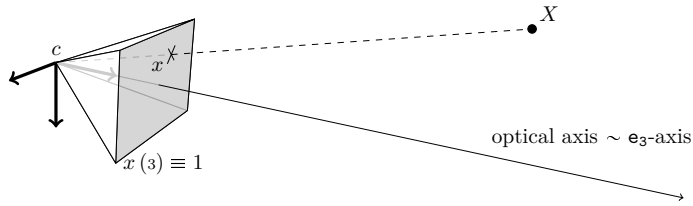


**Figure 2.2.** Projecting the 3-space point $X$ to the point $x$ on the image plane $\left\{ X \in \mathbb{R}^3 \mid \exists \alpha, \beta \in \mathbb{R} : X = (\alpha\ \beta\ \mathsf{e}_3) \right\}$. The optical axis corresponds to the $\mathsf{e}_3$-axis of the camera frame $c$. The $\mathsf{e}_1$-axis is directed to the right in view direction, the $\mathsf{e}_2$-axis down.

are introduced. In addition the **T**ri**F**ocal**T**ensor, used for modeling of stereo imaging in chapter 4.2.6, is described. For a more detailed description of this theoretical model see [HZ03] by Hartley et al. The definitions and derivations of the projective space and its entities, presented in this section, are adopted from their work.

### 2.3.1 Projection Model

The projection model of the pinhole camera is based on the assumption, that its physical process of imaging can be approximated by rays. These rays connect points in 3-space with their projection to the image. This is done by intersecting the ray passing through the 3-space point and the camera position, also called camera or projection center, with the image plane. The optical axis is given by the $\mathsf{e}_3$-axis of the camera frame $c$ and defines the image center. See figure 2.2 for a visualization of the evolved entities.

The camera plane is defined to be $\mathsf{e}_1$-/$\mathsf{e}_2$-plane in the camera frame. Correspondingly, the image plane is supposed to be the plane parallel to the camera plane, passing $(0, 0, 1)$. For a 3-space point $X = t^c_{cX}$, given in the camera coordinate frame $c$, the ray passing $t^c_{cX}$ and the camera center $t^c_{cc} = 0_3$ is given by the set:

$$\mathbf{r} \quad = \quad \left\{ r \in \mathbb{R}^3 \mid \exists \alpha \in \mathbb{R} : r = \alpha \cdot (t^c_{cX} - t^c_{cc}) \right\}$$

$$= \quad \left\{ r \in \mathbb{R}^3 \mid \exists \alpha \in \mathbb{R} : r = \alpha \cdot t^c_{cX} \right\} \tag{2.3.1}$$

Intersecting $\mathbf{r}$ with the image plane, is done by determining $\alpha$ in equation (2.3.1) such that:

$$\exists u, v \in \mathbb{R} \quad : \quad \alpha t^c_{cX} \; = \; \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \tag{2.3.2}$$

By this it follows that $\alpha = {t^c_{cX}}_{(3)}{}^{-1}$. The resulting vector

$$t^c_{cX'} \; = \; \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \; = \; \frac{1}{{t^c_{cX}}_{(3)}} \cdot t^c_{cX} \tag{2.3.3}$$

is the intersection with the image plane. Due to the fixed position in the $\mathsf{e}_3$-axis, $x = (u, v)$ can be defined to be the projection of $X$ by the camera $c$, having the image coordinates $u$,$v$.

When a point $X = t^a_{aX}$ is given in a coordinate frame $a$ other than the camera frame, a coordinate transformation has to be applied. This is done using the transformation $\Gamma^c_a$. The camera frame is defined by the cameras position $t^a_{ac}$, given by the camera center, and its orientation $\Phi^c_a$. According to equation (2.3.1) the ray in the camera frame is determined by:

$$\begin{aligned} \mathbf{r} \quad &= \quad \left\{ r \in \mathbb{R}^3 \mid \exists \alpha \in \mathbb{R} : r = \alpha \cdot t^c_{cX} \right\} \\ &= \quad \left\{ r \in \mathbb{R}^3 \mid \exists \alpha \in \mathbb{R} : r = \alpha \cdot \Gamma^c_a \left( t^a_{aX} \right) \right\} \\ &= \quad \left\{ r \in \mathbb{R}^3 \mid \exists \alpha \in \mathbb{R} : r = \alpha \cdot \left( \Phi^c_a \cdot (t^a_{aX} - t^a_{ac}) \right) \right\} \end{aligned} \tag{2.3.4}$$

Following that, normalizing $\Phi^c_a \cdot (t^a_{aX} - t^a_{ac})$ by its third component, results in the respective image coordinates $(u, v)^T$:

$$\begin{pmatrix} u \\ v \end{pmatrix} \; = \; \frac{\Phi^c_{a(1:2,:)} \cdot (t^a_{aX} - t^a_{ac})}{\Phi^c_{a(3,:)} \cdot (t^a_{aX} - t^a_{ac})} \tag{2.3.5}$$

It is worth noticing, that points having $\Phi^c_{a(3,:)} \cdot (t^a_{aX} - t^a_{ac}) = 0$ cannot be

projected to the image plane. This is due to the fact, that the according ray

$$\mathbf{r} = \left\{ r \in \mathbb{R}^3 \;\middle|\; r_{(3)} = 0 \;\wedge\; \exists \alpha \in \mathbb{R} : \begin{pmatrix} r_{(1)} \\ r_{(2)} \\ 0 \end{pmatrix} = \alpha \cdot t^c_{cX} \right\} \tag{2.3.6}$$

is resided in the camera plane parallel to the image plane. Thus, no point in $\mathbb{R}^3$ can be determined, being in $\mathbf{r}$ and the image plane at the same time.

### 2.3.2 Projective Geometry

The projective geometry arose from the Euclidean geometry by postulating the so called points at infinity. These provide the capability of intersecting parallel lines and planes, which is not possible for a Euclidean space. By this parallel hyper planes (i. e., lines in 2-space) intersect in sub-hyper planes (lines in 3-space, points in 2-space). This can be used to augment the basic pinhole model by vanishing points and other projective entities. See [HZ03] for a more comprehensive description. In [BS96] and [Art57] the projective space, being the foundation of projective geometry, is embedded to mathematical algebra.

Instead of intersecting rays with planes explicitly, as is done in section 2.3.1, the rays themselves are used for representation of projective entities. This is done by introducing the so called $\mathtt{n}$-dimensional projective space $\mathbb{P}^{\mathtt{n}}$. It consists of equivalence classes, called projective points. These are formed by the equivalence relation $\equiv$, on elements in $\mathbb{R}^{(\mathtt{n}+1)} \setminus \{0_{\mathtt{n}}\}$. Where:

$$\forall x, x' \in \mathbb{R}^{(\mathtt{n}+1)} \setminus \{0_{\mathtt{n}}\} : \qquad x \equiv x' \iff \exists\, \alpha \in \mathbb{R} \setminus \{0\} : x = \alpha x' \tag{2.3.7}$$

Note the similarity to the ray definition (2.3.4). By this $\mathbb{P}^{\mathtt{n}}$ is defined as:

$$\mathbb{P}^{\mathtt{n}} = \left\{ [x]_{\equiv} \;\middle|\; x \in \mathbb{R}^{\mathtt{n}+1} \setminus \{0_{\mathtt{n}+1}\} \right\}$$

In following the index $\equiv$ for the equivalence classes will be omitted since no other equivalence relation will be used. $\mathbb{P}^{\mathtt{n}}$ can be partitioned into the projective subspaces of homogeneous points $\mathbb{P}^{\mathtt{n}}_P$ and points at infinity $\mathbb{P}^{\mathtt{n}}_I$.

They are defined by:

$$\mathbb{P}^{\mathtt{n}}_P = \left\{ [x] \;\middle|\; x \in \mathbb{R}^{\mathtt{n}+1} \backslash \{0_{\mathtt{n}+1}\} \;\wedge\; x_{(\mathtt{n}+1)} \neq 0 \right\} \tag{2.3.8}$$

$$\mathbb{P}^{\mathtt{n}}_I = \left\{ [x] \;\middle|\; x \in \mathbb{R}^{\mathtt{n}+1} \backslash \{0_{\mathtt{n}+1}\} \;\wedge\; x_{(\mathtt{n}+1)} = 0 \right\} \tag{2.3.9}$$

Following equations (2.3.4) and (2.3.5), the projected image point can be identified with the according ray:

$$x = \begin{pmatrix} x_{(1)} \\ x_{(2)} \\ 1 \end{pmatrix} \in \mathbf{r} \quad \text{and thus:} \quad \mathbf{x} = [x] = \left[ \begin{pmatrix} x_{(1)} \\ x_{(2)} \\ 1 \end{pmatrix} \right] \tag{2.3.10}$$

Note that, in contrast to ray intersection (see equation (2.3.6)), this can be applied to rays in the camera plane (parallel to the image plane) also:

$$x = \begin{pmatrix} x_{(1)} \\ x_{(2)} \\ 0 \end{pmatrix} \in \mathbf{r} \quad \text{and thus:} \quad \mathbf{x} = [x] = \left[ \begin{pmatrix} x_{(1)} \\ x_{(2)} \\ 0 \end{pmatrix} \right] \tag{2.3.11}$$

Note further, that $0_3$ is in $\mathbf{r}$ but not included in $\mathbf{x}$. That is, the camera center is excluded from the interpretation of homogeneous points as rays. Equations (2.3.10) and (2.3.11) represent the two subspaces $\mathbb{P}^2_P$ and $\mathbb{P}^2_I$ respectively.

Beside the homogeneous points also homogeneous lines can be defined. Since it is a 2-space, a line has 2 degrees of freedom. Thus, a point in $\mathbb{P}^2$ can be used represent a line. This fact is known as point-line duality. Due to the different semantic of points and lines, the space of lines is called the dual space $\mathbb{P}^2_*$ of $\mathbb{P}^2$. Since these are dual but not identical spaces, a mapping has to be applied, when points and lines are related with each other. Since this is only a semantic mapping, it is omitted in the following. For two points $\mathbf{x}, \mathbf{x}' \in \mathbb{P}^2$ and two lines $\mathbf{l}, \mathbf{l}' \in \mathbb{P}^2_*$ holds:

$$\mathbf{x}^T \mathbf{l} = 0 \quad \Leftrightarrow \quad \mathbf{l} \quad \text{passes through} \quad \mathbf{x} \tag{2.3.12}$$

$$\mathbf{x} \neq \mathbf{x}' \quad \Leftrightarrow \quad \mathbf{l} := \mathbf{x} \times \mathbf{x}' \quad \text{defines a line in } \mathbb{P}^2_* \tag{2.3.13}$$

In (2.3.13) the inequality of points corresponds to the constraint, that the according rays in 3-space are not parallel. Note that this implies, that the

right hand equation is only defined for non parallel lines, since parallelism implies $\mathbf{x} \times \mathbf{x}' = [0_3]$ and $[0_3] \notin \mathbb{IP}^2$.

As for rays equivalence classes for linear mappings can be introduced. Therefore, a equivalence relation is augmented by:

$$\forall A, B \in \mathbb{R}^{\mathtt{n}\times\mathtt{m}} \backslash \{0_{\mathtt{n}\times\mathtt{m}}\} : \quad A \equiv B \Longleftrightarrow \exists \alpha \in \mathbb{R} \backslash \{0\} : A = \alpha B \tag{2.3.14}$$

**Projection** Using this notation and the equivalence classes from the projective space, the equations in section 2.3.1 can be simplified. That is, for a 3-space point $t^e_{eX}$ equation (2.3.5) and (2.3.2) can be replaced by:

$$\mathbf{x} = \left[ \begin{pmatrix} x(1) \\ x(2) \\ 1 \end{pmatrix} \right] = [\Phi^c_e \cdot (t^g_{eX} - t^e_{ec})] \tag{2.3.15}$$

$$= \left[ \left( \begin{array}{c|c} \Phi^c_e & -\Phi^c_e \cdot t^e_{ec} \end{array} \right) \cdot \begin{pmatrix} t^e_{eX} \\ 1 \end{pmatrix} \right] \tag{2.3.16}$$

$$= \left[ \left( \begin{array}{c|c} \Phi^c_e & -\Phi^c_e \cdot t^e_{ec} \end{array} \right) \right] \cdot \left[ \begin{pmatrix} t^e_{eX} \\ 1 \end{pmatrix} \right] =: \mathbf{PX} \tag{2.3.17}$$

Where $\mathbf{P}$ is the projection mapping (camera for short) for the camera $c$ with respect to the global frame $e$ and $\mathbf{X}$ the homogeneous point representing $t^e_{eX}$. To ease the readability, $\mathbf{P}$ and $c$ are identified with each other in the following. Moreover, by writing $\mathbf{P}^c$ refers to the camera induced by $\Gamma^c_e$. Due to the scale invariance of all involved entities, this equation can be applied to any representants of $\mathbf{P}$ and $\mathbf{X}$. That is:

$$\forall P \in \mathbf{P}, X \in \mathbf{X} \ : \ PX \in \mathbf{x}$$

In most cases the so called camera matrix $K$ is included in the projection. It combines scalings $f_{\mathtt{e}_1}, f_{\mathtt{e}_2}$, a skew $\alpha$ and a translation $p$ in the image plane. They describe the cameras focal lengths, non perpendicular image axis and the image center respectively (see [HZ03], [Bou99]). It has to be stated, that focal lengths in this model do not correspond directly to focal lengths in physical cameras. They rather describe the distance between the center of projection and the image plane (i. e., the distance between camera

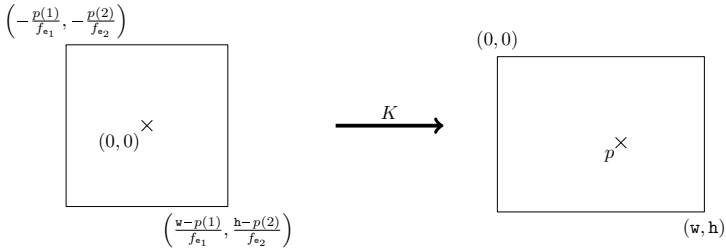


**Figure 2.3.** $K$-matrix mapping for a camera with image size of $\mathtt{w} \times \mathtt{h}$ pixels, its focal lengths $f_{\mathtt{e}_1}, f_{\mathtt{e}_2}$ and image center $p$.

and image plane). The skew is neglected in general and won't be discussed further. The image center $p$ denotes the intersection point of the cameras optical axis and the image plane. By this the camera matrix $K$ is:

$$K = \begin{pmatrix} f_{\mathtt{e}_1} & \alpha & p(1) \\ 0 & f_{\mathtt{e}_2} & p(2) \\ 0 & 0 & 1 \end{pmatrix} \tag{2.3.18}$$

This augments the projective camera by the respective properties of an ideal physical pinhole camera. It has to be a stated, that, in contrast to real cameras, the focal lengths and $p$ are given in pixels mostly. By this the transformation is independent of the physical dimensions of a camera. In especially, it transfers the projected points to pixel positions in the image, instead of absolute positions on the camera chip. In figure 2.3 an exemplary $K$-matrix mapping is visualized.

For a camera $c$ with the global pose $\Gamma_e^c$ and a camera matrix $K$, the general projection matrix $\mathbf{P}$ ends up with:

$$\mathbf{P} = [K] \circ \Gamma_e^c = [K \cdot (\Phi_e^c | - \Phi_e^c \cdot t_{ec}^e)] \tag{2.3.19}$$

**Homogenization** To relate the imaging process via projective space and the computation of $u$, $v$ in section 2.3.1 the so called homogenization is needed. That is, for the homogeneous point $\mathbf{x}$ a normalization, according to equation (2.3.5) on page 17, has to be carried out. To do so, an arbitrary

representative $x \in \mathbf{x}$ is chosen and divided by its third component.

$$\begin{pmatrix} u \\ v \end{pmatrix} = \mathbf{H}(\mathbf{x}) = \frac{1}{x(3)} \cdot \begin{pmatrix} x(1) \\ x(2) \end{pmatrix} \tag{2.3.20}$$

Throughout this thesis the mapping $\mathbf{H}$ will be referred to as homogenization. By dividing the vector $x$ by its third component, the scale ambiguity from the equivalence class $\mathbf{x}$ is removed. Thus, $\mathbf{H}(\cdot)$ is a bijective mapping between $\mathbb{P}^2_P$ and $\mathbb{R}^2$.

**Fundamental Matrix** An other important projective mapping is the fundamental matrix $\mathbf{F}$, for a detailed description see [HZ03]. Consider two cameras $\mathbf{P}$, $\mathbf{P}'$ with different camera centers and a point $\mathbf{X} \in \mathbb{P}^3$. The fundamental matrix can be computed from $\mathbf{P}$ and $\mathbf{P}'$ and relates image points $\mathbf{x}$ and $\mathbf{x}'$ in the first and second camera respectively. That is:

$$\mathbf{x} = \mathbf{P} \cdot \mathbf{X} \quad \wedge \quad \mathbf{x}' = \mathbf{P}' \cdot \mathbf{X} \qquad \Longrightarrow \qquad \mathbf{x}'^T \cdot \mathbf{F} \cdot \mathbf{x} = 0 \tag{2.3.21}$$

For cameras having certain spatial relations (see [HZ03]) this mapping can be used, to correlate points and lines in both images. I. e., the projection of a ray from one camera results in a line in the second image. Let $\mathbf{l}' = \mathbf{F} \cdot \mathbf{x}$. By this the following holds:

$$\mathbf{x}'^T \cdot \mathbf{F} \cdot \mathbf{x} = 0 \iff \mathbf{x}'^T \cdot \mathbf{l}' = 0 \overset{(2.3.12)}{\iff} \mathbf{l}' \text{ intersects } \mathbf{x}' \tag{2.3.22}$$

Note that points $\mathbf{x}''$ exist, fulfilling $\mathbf{x}''^T \cdot \mathbf{l}' = 0$ without being correspondences of $\mathbf{x}$. Thus, $\mathbf{l}'$ is the subspace in the second cameras image, that includes al points $\mathbf{x}'$ potentially corresponding to $\mathbf{x}$. This line is also called the **epipolar line** of $\mathbf{x}$ with respect to $\mathbf{P}$ and $\mathbf{P}'$.

It has to be stated, that $\mathbf{F}$ maps between the dual spaces $\mathbb{P}^2_*$ and $\mathbb{P}^2$. Thus, this mapping is a tensor, but not a matrix, i. e., no linear mapping. Since the interpretation of homogeneous mappings as matrices over projective spaces is sufficient for this thesis and to ease the readability, this imprecision is neglected in the following.

### 2.3.3 Stereo Geometry

The stereo geometry makes use of two rigidly coupled cameras, that have an overlapping field of view. The known coordinate transformation $\Gamma_c^{c'}$ from the master $c$ to the slave camera $c'$ can be used, to relate a pixel in one camera to a line the other camera (see previous section).

For a standard stereo camera rig $\Gamma_c^{c'}$ is assumed, to be a simple translation along the axis, that corresponds to the image width (i. e., $\mathsf{e}_1$-axis). This 1-dimensional translation is referred to as baseline $b$. Thus:

$$\Gamma_c^{c'} \quad = \quad \left\langle \Phi_c^{c'}, t_{cc'}^c \right\rangle \quad = \quad \left\langle 1^3, \begin{pmatrix} b \\ 0 \\ 0 \end{pmatrix} \right\rangle \tag{2.3.23}$$

Due to the overlapping field of view and $b > 0$ the camera setup is known to fulfill the requirements of equation (2.3.22). Moreover, the simple transformation along $\mathsf{e}_1$ allows for relating the image rows in both cameras. Let $\mathbf{F}$ the fundamental matrix resulting from $\Gamma_c^{c'}$. By this it holds:

$$\forall\, \mathbf{x} \in \mathbb{P}^2 \;: \quad \left[ \begin{pmatrix} 0 \\ l_2 \\ l_3 \end{pmatrix} \right]^T \cdot \mathbf{x} = 0 \quad \Longleftrightarrow \quad \left[ \begin{pmatrix} 0 \\ l_2 \\ l_3 \end{pmatrix} \right] = \mathbf{F} \cdot \mathbf{x} \tag{2.3.24}$$

By this it follows, that two corresponding image points $\mathbf{x} \leftrightarrow \mathbf{x}'$ lie on the same line $\mathbf{l} = \mathbf{l}'$ in the master and slave camera respectively. Moreover, $\mathbf{l} = \mathbf{l}'$ are parallel to the axis in the image width ($l_1 = 0$). Thus, a point $\mathbf{X}$ seen from both cameras, projects to the same image row in the master and slave image. This implies, that $\mathbf{x} = [x]$ and $\mathbf{x}' = [x']$ differ only in a displacement $d$ along the image rows, i. e., $[x] = [x' + \left(\begin{smallmatrix} d \\ 0 \end{smallmatrix}\right)]$. The displacement $d$ is known as disparity and has the same sign as $b$, i. e., $d > 0$. See figure 2.4 for a visualization of the involved entities.

### 2.3.4 TriFocal Tensor

In contrast to the fundamental matrix describing the properties of a combination of two cameras, the **T**ri**F**ocal **T**ensors describe the relationships in a triplet of cameras $\mathbf{P}$, $\mathbf{P}'$ and $\mathbf{P}''$. Assuming a point $\mathbf{X}$ is visible in all three cameras, it projects to three corresponding entities. For each combination of entities to relate (3 points, 2 points with 1 line etc.), a different tensor

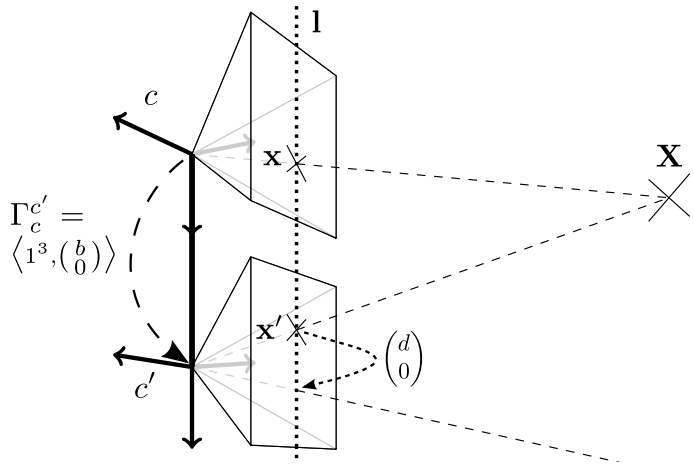


**Figure 2.4.** Basic structure for a standard stereo rig. The baseline $b$ denotes the 1-dimensional master-slave translation $t_{cc'}^{c} = (b\ 0\ 0)$, $d$ the disparity between the stereo correspondences $x$ and $y$ as projections of $X$.

notation results. In the following, $\mathcal{T}$ denotes the tensor relating a point, a line and a point, using the given cameras. That is:

$$\mathcal{T} \ : \ \mathbb{P}^2 \times \mathbb{P}_*^2 \times \mathbb{P}^2 \ \mapsto \ \mathbb{R}$$

Let $\mathbf{x}$ and $\mathbf{x}''$ two corresponding points along with a corresponding line $\mathbf{l}'$ in the respective cameras. The line correspondence $\mathbf{l}'$ is defined by a line containing $\mathbf{x}' = \mathbf{P}'\mathbf{X}$. To be a non degenerated correspondence, $\mathbf{l}'$ must not equal $\mathbf{F}\mathbf{x}$, i. e., the line of all possible correspondences to $\mathbf{x}$ in $\mathbf{P}'$. In the following, it is assumed, that the cameras are in no degenerated relation. For a detailed description of TFTs, and their properties and restrictions see [HZ03].

Following [HZ03], the TFT $\mathcal{T}$ can be used, to map the correspondences $\mathbf{x}$ and $\mathbf{l}'$ in $\mathbf{P}$ and $\mathbf{P}'$ respectively to their analogon $\mathbf{x}''$ in $\mathbf{P}''$. To depict this process, consider the plane $\pi'$, resulting from un-projecting $\mathbf{l}'$ to 3-space (see figure 2.5). From the intersection of $\pi'$ and the 3-space view ray of $\mathbf{x}$ the three space point $\mathbf{X}$ arises. Projecting this to the third camera, gives

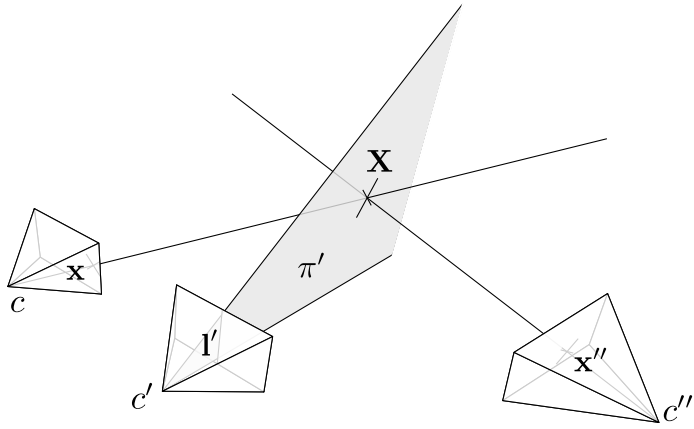


**Figure 2.5.** Visualization of TFT mapping. Point-line correspondence $\mathbf{x} \leftrightarrow \mathbf{l}'$ in $c$ and $c'$ respectively to $\mathbf{x}''$ in $c''$. Mapping via the back projected plane $\pi'$ for $\mathbf{l}'$. $\mathbf{X}$ marks corresponding 3-space point.

the sought-after correspondence $\mathbf{x}''$.

The TFT can be computed from the three projection matrices, assuming $\mathbf{P} = \left( \begin{array}{c|c} \mathbf{I} & 0_3 \end{array} \right)$ without loss of generality. Using

$$\begin{aligned} \mathbf{P}' &= \left[ \begin{pmatrix} a_1^1 & \cdots & a_4^1 \\ \vdots & \vdots & \vdots \\ a_1^3 & \cdots & a_4^3 \end{pmatrix} \right] \\ \mathbf{P}'' &= \left[ \begin{pmatrix} b_1^1 & \cdots & b_4^1 \\ \vdots & \vdots & \vdots \\ b_1^3 & \cdots & b_4^3 \end{pmatrix} \right] \end{aligned}$$

the TFT $\mathcal{T}$ mapping from $\mathbf{P}$, $\mathbf{P}'$ to $\mathbf{P}''$ results in:

$$\mathcal{T}_{\mathtt{i}}^{\mathtt{jk}} = a_{\mathtt{i}}^{\mathtt{j}} \cdot b_4^{\mathtt{k}} - a_4^{\mathtt{j}} \cdot b_{\mathtt{i}}^{\mathtt{k}} \tag{2.3.25}$$

This equation is to be evaluated by Einstein's Sum Convention (see [BS96]),

i. e., by summation over all non fixed indices. Using $x$, $l'$ and $x''$ as representatives for the respective projective entity, it holds:

$$x''_{(\mathtt{k})} \quad = \quad l'_{(\mathtt{j})} x_{(\mathtt{i})} \mathcal{T}_{\mathtt{i}}^{\mathtt{jk}} \tag{2.3.26}$$

To ease the readability, the tensor notation $x^{\mathtt{i}}$ and $l_{\mathtt{j}}$ for points and lines is substituted by $x_{(\mathtt{i})}$ and $l_{(\mathtt{j})}$ respectively. It is worth mentioning that this mapping is performed without explicite triangulation of $\mathbf{X}$. By this the TFT mapping is much more robust to noisy correspondences compared to triangulation and reprojection.

## 2.4 Sensing Devices

Sensing devices are the basis for every SLAM system. They provide information on the systems conditions, environment or movement. This way, the system percepts its state and its environment, enabling it, to simultaniously estimate its pose and environment map. The sensors used for SLAM methods analyzed in this thesis, are cameras and inertial measurement units. The properties of these sensing devices are discussed in the following two sections.

### 2.4.1 Digital Cameras

In section 2.3 the theoretic pinhole model for the imaging process in cameras was described. Since this covers only an ideal camera, additional methods have to be used, to allow for its application to physical cameras. Due to imperfect production processes and the approximative nature of the pinhole model, non linear model errors arise. These cannot be covered by the camera calibration matrix $K$.

Among others, such errors are the chromatic aberration and vignetting (see [Sze11]), effecting the images (chromatic-) illumination. In the following, it is assumed, that cameras are used, that allow for neglecting these effects in the used algorithms. More important to visual SLAM are the geometric imaging errors.

### Lense Distortion

A major violation of the ideal image formation model is the geometric image deformation. Most of it is caused by the lense distortion. It cannot be neglected in visual sensing, especially when low cost lenses are used. It is mainly caused by the imperfect production processes. It becomes noticeable, when the images of straight lines are no longer straight, resulting in curvatures of the captured images.

The model, used in the proposed SLAM system for compensating this distortion, was introduced by Heikkila (see [HS97]). It uses polynomial mappings $p^{\mathtt{r}}$ and $p^{\mathtt{t}}$ to compensate for the radial and tangential distortion respectively. Therefore, each pixel pixel $x$ is scaled, using $p^{\mathtt{r}}$ applied to the radius $r$ (distance of $x$ to distortion center). Afterwards $p^{\mathtt{t}}$ is used, to determine an offset for $x{\cdot}p^{\mathtt{r}}{}_{(r)}$. Due to the symmetry of the distortion model, the polynomials can be estimated, to either map from the distorted to the undistorted image (undistortion) or vice versa (distortion). The camera calibrations for the dataset discussed in section 7.2 (page 170) are provided in the respective format. Thus, this algorithm was found, to be the most straight forward method, to be integrated in the developed SLAM system.

Mostly, the backward mapping (inverse distortion) is used, for undistortion of images. That is, the pixel $x$ in the undistorted image is mapped to an image coordinate $x^o$ in the original image. The image value of $x$ is then determined, by interpolating the values of the pixels, neighboring $x^o$. By this wholes in the undistorted image are avoided, that arise from integral pixel mapping.

### Stereo Camera Systems

In addition to the lense distortion, a misalignment between the two stereo cameras has to be considered. Neglecting this transformation, results in violations of the geometric prerequisites for stereo vision (e. g., point to line mapping, see section 2.3.3, page 23). Thus, it can be seen as geometric imaging error.

As described in section 2.3.3, the transformation between both cameras $\Gamma_c^{c'} = \left\langle \Phi_c^{c'}, t_{cc'}^c \right\rangle$ is assumed to be a pure translation. That is, the rotation $\Phi_c^{c'} = 1^3$ and the translation $t_{cc'}^c = (\,b\;0\;0\,)$ for the baseline $b$. Obviously,

the baseline $b$ is dependent on the camera rig and has to be determined. Moreover, the true alignment deviates from the theoretical assumption. This deviation has to be determined and compensated, when considering real stereo camera setups.

Fusiello et al. (see [FTV00]) proposed a method to fully rectify stereo image pairs. Therefore, the true stereo calibration $\Gamma_{\gamma}^{\gamma'}$ is assumed to be known and the images to be undistorted. Using $t_{\gamma\gamma'}^{\gamma}$, a rotation $\Phi_{\gamma}^{c}$ is computed, satisfying

$$\Phi_{\gamma}^{c} \cdot t_{\gamma\gamma'}^{\gamma} = \begin{pmatrix} \left| t_{\gamma\gamma'}^{\gamma} \right|_2 \\ 0 \\ 0 \end{pmatrix} = t_{cc'}^{c} = \begin{pmatrix} b \\ 0 \\ 0 \end{pmatrix} \tag{2.4.1}$$

Note that the above constraint is invariant to rotation around the $\mathbf{e}_1$-axis. This is handled, by fixing this rotation to half way between the orientations of $\gamma$ and $\gamma'$ respectively. Using $\Phi_{\gamma}^{c}$ and $\Phi_{c}^{c'}$, two new virtual cameras can be determined, fulfilling the stereo constraints. The images are transformed to the virtual cameras by rotating the image planes accordingly (for a pixel $x$ rotating the ray $\mathbf{x}$). Finally for each pixel $x$ in the rectified images, a backward mapping is applied (see image undistortion).

### Pixel Normalization

Throughout this thesis, all pixel coordinates refer to normalized pixels. That is, the undistortion, rectification and the application of the inverse camera calibration matrix $K^{-1}$ are assumed. The camera calibrations (calibration matrix, undistortion polynomials and stereo rectification) are determined, using the software framework [Sch] described in [SBK08].

Since these transformations of images are time consuming, not the whole image is normalized for the proposed algorithms. Only the pixel positions, used for visual perception of the SLAM system (see section 2.5), are mapped. These positions of interest in the images are scaled, translated (undistortion and inverse $K$) and rotated (rectification). This approach reduces the computational effort, since only a small number of image coordinates have to be transformed.

### 2.4.2 Inertial Measurement Units

**I**nertial **M**easurement **U**nits (IMU) provide information on the systems acceleration and rates of turn. These can be used to determine position and orientation increments by applying strap down navigation. The most complex IMUs, used for inertial navigation systems, provide measurements with high precision and sample rates. They are priced tens to hundred thousands of Euros and the precision is sufficient even for long term navigation. Due to their high costs, their application area is limited to navigation for large ships or airplanes.

With the ongoing development of integrated circuits, IMUs are available with decreasing size and power consumption and at low costs. Nowadays, simple IMUs are integrated in computer game controlers, cell phones and the like, and provide intuitive controls for games and other applications. Since such simple sensors suffer from high noise and sensitivity to environmental influences (temperature, electrical interference etc.), more sophisticated IMUs have been developed. With a tradeoff in price they are more stable and precise, allowing for short term navigation tasks, especially for orientation estimation. Because they are priced at a few hundreds of Euros, they are helpful in aiding SLAM systems based on vision sensors. The acceleration and turn rate measurements can be used directly, to predict the systems movement for a short time. When new visual measurements are provided, the prediction of the systems pose and other parameters can be improved in accuracy and stability.

#### Sensor Hardware

Nowadays various types of sensors for measuring inertial entities are available. A well known high precision turn rate sensor is the ring laser gyroscope. It makes use of two standing laser waves, being trapped in a polygon of mirrors. Rotating the system influences the distance the lasers rays must travel, resulting in interference. This effect is measured and provides information on the occurred rate of turn. For acceleration measurements similar systems are available. Nevertheless, the sensor class with the highest quantity are the **M**icro **E**lectro-**M**echanical **S**ystems, MEMS for short. For an exhaustive description of such sensor systems and their combinations see [TW04].

Because measurements acquired by IMUs undergo certain disturbances, these have to be taken into account. The main errors are incorporated by biases and scale factors (see [Wen07]). For short term integration of inertial data these can be modeled using constants, even though they are influenced by certain environment conditions. Other error sources like axis misalignment and non linear sensor response are complex to model. Most often they are partially pre-calibrated and implicitly compensated, when the sensor measurements are delivered by device drivers. Due to the small impact of the remaining compensation errors to short term integration, these are not considered in this thesis.

In addition to such systematical errors, that can be compensated during runtime, hardware limitations imply constraints to the navigating system. On the one hand, environmental conditions have to be taken into account. That is, the sensor delivers reliable data only for certain temperature and barometric pressure ranges, what constraints their applicability. On the other hand, measurement ranges (scales) have to be considered. For very small accelerations and turn rates, the sensors sensitivity is not sufficient to resolve the movement. This is caused by low signal to noise ratios, corse quantization in analog-digital conversion or mechanical limitations. Due to this, such small variations cannot be detected, what is known as hysteresis. When accelerations or turn rates exceed certain limits, unpredictable behavior occurs, resulting in erroneous observations. This is due to the fact, that the measuring devices are linear only within their scales. Moreover, value ranges for discretization have to be assumed and mechanical limits apply.

#### Strap Down Navigation

Strap Down Navigation describes the method of incrementally determining the systems position and orientation, using measurements of accelerations and rates of turn. The accelerations and rates of turn are usually provided by inertial measurement units. The position results from double integration of acceleration, and the orientation is computed by integrating the turn rates. Thus, the orientation estimates are much more precise and stable over time. The models used in this thesis are simplifications of the full models, discussed by Wendel in [Wen07]. Due to the low precision of the IMU used for the proposed hardware setup, the impact of the remaining model errors

to the estimation process are comparably small.

**Orientation** For integrating the turn rates, incremental rotation matrices (see 2.2.1) are used. Since the systems rates of turn $w^i_{ni}$ are assumed to be constant for the IMUs measurement interval, first order Taylor expansion (see [BS96]) can be used for numeric integration. Therefore, the rotations derivative in time domain $\tau$ has to be determined for the linearization point $\tau_0$. Let $\Phi^n_{i(\tau)} = \{R^n_i, \phi_{(\tau)}\}$ the systems time dependent rotation as well as

$$\phi_{(\tau_0)} \;=\; 0 \qquad \text{and} \qquad \left.\frac{\partial \phi_{(\tau)}}{\partial \tau}\right|_{\tau_0} \;=\; w^i_{ni(\tau_0)} \;=:\; \begin{pmatrix} w_1 \\ w_2 \\ w_3 \end{pmatrix}$$

By this the rotations derivative is determined by:

$$\begin{aligned}
\left.\frac{\partial \Phi^n_i}{\partial \tau}\right|_{\tau_0} &= \left.\frac{\partial \{R^n_i, \phi_{(\tau)}\}}{\partial \tau}\right|_{\tau_0} = R^n_i \cdot \left.\frac{\partial R_{(\phi(\tau))}}{\partial \tau}\right|_{\tau_0} \\
&= R^n_i \cdot \sum_{\mathtt{i}=1}^{3} \left( \left.\frac{\partial R_{(\phi)}}{\partial \phi_{(\mathtt{i})}}\right|_{\phi(\mathtt{i},\tau_0)} \cdot \left.\frac{\partial \phi_{(\mathtt{i},\tau)}}{\partial \tau}\right|_{\tau_0} \right) \\
&= R^n_i \cdot \sum_{\mathtt{i}=1}^{3} \left( \left.\frac{\partial R_{(\phi)}}{\partial \phi_{(\mathtt{i})}}\right|_{0} \cdot w^i_{ni(\mathtt{i},\tau_0)} \right) \\
&= R^n_i \cdot \left( \begin{pmatrix} 0&0&0 \\ 0&0&-1 \\ 0&1&0 \end{pmatrix} \cdot w_1 + \begin{pmatrix} 0&0&1 \\ 0&0&0 \\ -1&0&0 \end{pmatrix} \cdot w_2 + \begin{pmatrix} 0&-1&0 \\ 1&0&0 \\ 0&0&0 \end{pmatrix} \cdot w_3 \right) \\
&= R^n_i \cdot \begin{pmatrix} 0 & -w_3 & w_2 \\ w_3 & 0 & -w_1 \\ -w_2 & w_1 & 0 \end{pmatrix} \qquad (2.4.2)
\end{aligned}$$

Thus, using the first order Taylor expansion, the global orientation integration step of length $\delta\tau$ for time $\tau_0$ is described by the incremental orientation as:

$$\begin{aligned}
\Phi^n_{i(\tau_0+\delta\tau)} &\approx \Phi^n_{i(\tau_0)} + \delta\tau \cdot \frac{\partial \Phi^n_{i(\tau_0)}}{\partial \tau} \\
&= \Phi^n_{i(\tau_0)} \cdot \left( 1^3 + \delta\tau \cdot \begin{pmatrix} 0 & -w_3 & w_2 \\ w_3 & 0 & -w_1 \\ -w_2 & w_1 & 0 \end{pmatrix} \right) \qquad (2.4.3)
\end{aligned}$$

$$\begin{aligned} &\overset{(*)}{\approx} && \Phi^n_{i\,(\tau_0)} \cdot R\big(\delta\tau \cdot w^i_{ni\,(\tau_0)}\big) \\ &= && \left\{R^n_{i\,(\tau_0)}, \delta\tau \cdot w^i_{ni\,(\tau_0)}\right\} \quad = \quad \Phi^n_i + \delta\tau \cdot w^i_{ni\,(\tau_0)} \end{aligned} \tag{2.4.4}$$

Equality $(*)$ is based on the assumption, that for small angles $\alpha$ the approximations $\sin(\alpha) \approx \alpha$ and $\cos(\alpha) \approx 1$ hold. It has to be applied, since the norm of the matrix, resulting from (2.4.3), is $\neq 1$. By this it is no rotation matrix. The deviations for the angle approximation are:

$$\sin(0.1^\circ) - \frac{0.1 \cdot \pi}{180} \approx 1e-9 \quad \text{and} \quad \cos(0.1^\circ) - 1 \approx 1.5e-6 \tag{2.4.5}$$

$$\sin(1^\circ) - \frac{1 \cdot \pi}{180} \approx 1e-6 \quad \text{and} \quad \cos(1^\circ) - 1 \approx 1.5e-4 \tag{2.4.6}$$

As can be seen, the error of this approximation increases strongly with increasing $\alpha$. Thus, it is important, that the integration is done with a high update rate.

As mentioned above, the IMUs output data is biased and scaled. Thus, correction terms $b_w$ (3-vector) and $s_w$ (scalar) are introduced for the bias and the scale respectively, to compute the $w^i_{ni}$ from the measured turn rates $w$. By this the new incremental rotation $\Phi^n_{i\,(\tau_0+\delta\tau)}$ is determined using $\Phi^n_{i\,(\tau_0)} = \{R^n_i, 0\}$:

$$\Phi^i_{n\,(\tau_0+\delta\tau)} \quad = \quad \Phi^i_{n\,(\tau_0)} + \delta\tau \cdot (s_w \cdot w + b_w) \tag{2.4.7}$$

As mentioned above, this is a simplification of the numerical integration discussed in [Wen07]. In especially the compensation of the earth's turn rate is omitted. This is due to the fact, that the rotation between the earth coordinate frame $\varepsilon$ and the used reference system $e$ is not known completely (unknown $\phi_\gamma$ in section 2.2.2). Thus, also the earths rates of turn are not known in $e$.

**Position and velocity** The IMUs accelerometers not only measure the systems acceleration with respect to the earth frame but also the earths gravity. Thus, the integration of velocities and positions is more complex. The systems horizontal orientation relative to the earth surface has to be used, to relate the measured acceleration and the gravity vector $\gamma$. By this it can

be used to exclude $\gamma$ from the accelerometer measurements. Note that this requires the global (i. e., environment) frame $e$ to be aligned with the earths north/east-plane (see section 2.2.2). By this the sensors initial coordinate system cannot be used as absolute reference frame.

Let $a = (\, a_1 \; a_2 \; a_3 \,)$ the accelerations measured by the IMU for the respective axis. To compute the IMUs acceleration $a^i_{ni}$ in its body fixed coordinate frame $i$, the biases $b_a$, scales $s_a$ and the earths gravity $\gamma$ have to be compensated. That is:

$$a^i_{ni} \quad = \quad s_a \cdot a + b_a + \gamma^i \tag{2.4.8}$$

By this the IMUs acceleration in the navigation frame $n$ is given by:

$$a^n_{ni} \quad = \quad R^n_i \cdot (s_a \cdot a + b_a) + R^n_e \cdot \gamma^e \tag{2.4.9}$$

In contrast to the orientation, the position is given by double integration of acceleration. Thus, the second order Taylor expansion has to be applied. Keeping in mind, that $a$ is assumed constant, the first order Taylor expansion is sufficient for the propagation of velocity in time. By this the time increments for $t^n_{ni}$ and $v^n_{ni}$ respectively can be approximated. Using $\tau_0$ as linearization point for integration of length $\delta\tau$ results in:

$$\begin{aligned}
t^n_{ni(\tau_0+\delta\tau)} &= t^n_{ni(\tau_0)} + \delta\tau \cdot \left.\frac{\partial t^n_{ni}}{\partial \tau}\right|_{\tau_0} + \frac{1}{2} \cdot \delta\tau^2 \cdot \left.\frac{\partial^2 t^n_{ni}}{\partial^2 \tau}\right|_{\tau_0} && (2.4.10)\\
&= t^n_{ni(\tau_0)} + \delta\tau \cdot v^n_{ni(\tau_0)} + \frac{1}{2} \cdot \delta\tau^2 \cdot a^n_{ni(\tau_0)} && (2.4.11)\\
&= t^n_{ni(\tau_0)} + \delta\tau \cdot v^n_{ni(\tau_0)} \\
&\quad + \frac{1}{2} \cdot \delta\tau^2 \cdot (R^n_{i(\tau_0)} \cdot (s_a \cdot a + b_a) + R^n_{e(\tau_0)} \cdot \gamma^e) && (2.4.12)
\end{aligned}$$

$$\begin{aligned}
v^n_{ni(\tau_0+\delta\tau)} &= v^n_{ni(\tau_0)} + \delta\tau \cdot \left.\frac{\partial v^n_{ni}}{\partial \tau}\right|_{\tau_0} && (2.4.13)\\
&= v^n_{ni(\tau_0)} + \delta\tau \cdot a^n_{ni(\tau_0)} && (2.4.14)\\
&= v^n_{ni(\tau_0)} + \delta\tau \cdot (R^n_{i(\tau_0)} \cdot (s_a \cdot a + b_a) + R^n_{e(\tau_0)} \cdot \gamma^e) && (2.4.15)
\end{aligned}$$

As for the orientation integration this is a simplification of the model proposed in [Wen07]. This simplification is applied, since some model violations can be neglected, due to the low precision of the IMU and the small navigation area (typical not more then a few kilometers), and other side effects, as the earths turn rate, cannot be compensated.

## 2.5 Feature Matching

When visual measurements shall be provided by image processing, relations between images have to be determined. For the SLAM system, proposed in this thesis, these relations are given by feature matching. Features are defined to be small patches in an image, that are especially distinctive. The process of feature matching correlates features in both images, that are likely, to correspond to the same entity in the observed scene. Automated methods for solving this problem are studied since the beginning of digital image processing.

The most simple methods use windows of fixed size (patches) to match similar patches of same size in the second image. This is done, comparing all possible patches from one image with all possible patches in the other image. The comparison is done, using some kind of matrix norm on the intensity difference of the respective image patches. Examples for such norms are the mean of absolute differences (MAD) or the sum of squared differences (SSD). To improve the robustness to intensity changes, the normalized cross correlation (NCC) was introduced (see [Sze11]).

Other methods detect image patches, that are likely to be more distinctive than other patches. They use the so called auto-correlation (see [Jäh02], [Sze11]), as is done for the "KLT"-features (see [ST94], [BM04]) or the Harris-corners [HS88]. They are named after **K**anade, **L**ucas and **T**omasi, who proposed the foundation of the respective tracking methods. These patches can be matched using the equality measures above, but also provide their own specialized methods.

Since the above methods use fixed window sizes, they are robust neither to rotational nor to scale changes. Even though improvements for these methods exist (see [MS04]), more sophisticated algorithms have been developed. These analyze the structure of the image and provide features, that

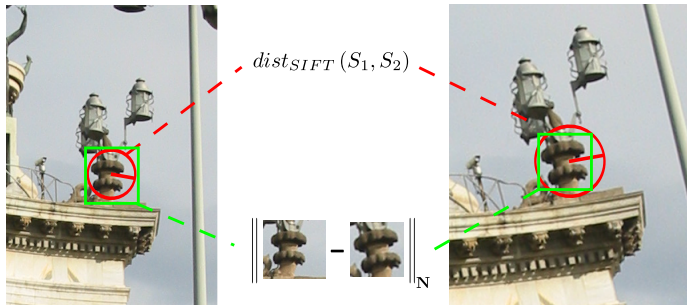


**Figure 2.6.** Comparison of window based and SIFT feature matching. Green: The fixed size window matcher, using a standard norm $N$, such as MAD, SSD etc.. Note the visual similarity in the window blocks, that is not detectable by image differencing. Red: The SIFT descriptor matching, compensating for rotation and scale of the feature, using the rotation and scale invariant descriptors visualized.

can vary amongst others in their size in the image. One example are the **M**aximally **S**tabel **E**xtremal **R**egions (MSER, see [Mat+02], [DB06]).

The feature detection and matching method, used in the SLAM filter implementation, proposed in this thesis, is based on the famous **S**cale **I**nvariant **F**eature **T**ransform (SIFT, see [Low04]). The features, detected by this method, have proven to be robust to illumination, rotation and scale changes. This way, SIFT outperforms window based algorithms, as is visualized in figure 2.6. Even though improved specializations have been discussed (see [YM09]), SIFT was chosen, due to the availability of fast (i. e., real-time capable) implementations.

The software library SiftGPU (see [Wu07]) used here, implements the complete SIFT algorithm on GPU (processing units of modern graphic cards). This implementation has proven to be well suited to real-time requirements made for the proposed SLAM system. This is due to the fact, that it allows for a high frame-rate of detection and matching. I. e., because the feature matching has to be performed on three images at once (old master to new master to new slave image) this high performance is needed. Moreover, it provides a high reliability in time effort. That is, the variance on the frame-rate is rather small compared to the average

frame-rate, avoiding lacks due to computation time peaks. Finally, beside the feature matches it also provides the full SIFT-descriptors (see [Low04] for a detailed description). These are used as additional information for the environment map database.

Chapter 3

# Linear Probabilistic Estimation

## 3.1 Probability Theory

The probability theory deals with the modeling of random events, variables and functions. These entities are used, to describe non deterministic events and processes, or those for which the rules driving them are unknown. Flipping a coin or rolling a dice for example undergo the rules of physics. In general not all constraints controlling such processes are known. Thus, the result cannot be predicted. To handle such problems, the probability theory uses mathematical models to give a notion of how likely the occurrence of a special result or a set of results is.

The definitions and conclusion discussed in this section are taken from [BS96]. For more detailed and exhaustive elaboration of terms and concepts see [RS01] and [Çin10].

### 3.1.1 Elementary Stochastic

An elementary event $\{e_i\}$ (a single element set, $e_i$ for short) is defined as the result of a random experiment, e. g., 1 when throwing a dice. Let $\mathcal{E} = \{e_1 \dots e_n\}$ a finite set of elementary events, where all $\{e_i\}$ are equiprobable. The likelihood of $e_i$ or an event $A \subseteq \mathcal{E}$ is represented by a real number $0 \leqslant p \leqslant 1$. A likelihood of $p = 0$ represents the impossible event, and $p = 1$ the inescapable event.

Let $E = 2^{\mathcal{E}}$. Using a mapping $\mathcal{P} : E \to [0, 1]$ every event is mapped to a likelihood. This mapping is defined by:

$$\mathcal{P}(A) = \frac{|A|}{|\mathcal{E}|} \quad \forall A \in \mathcal{E}$$

For example when rolling a 6-sided dice, it holds $\mathcal{E} = \{1 \dots 6\}$. That is, for the likelihood of the outcome of 2 or 4:

$$\mathcal{P}(\{2,4\}) = \frac{|\{2,4\}|}{|\mathcal{E}|} = \frac{2}{6}$$

To generalize this notion of probability to non finite sets $\mathcal{E}$, *Kolmogorows axioms of calculus of probabilities* are used. Let $\zeta \subseteq E := 2^{\mathcal{E}}$ such that:

$$1. \qquad \varnothing \in \zeta \quad \wedge \quad \mathcal{E} \in \zeta \tag{3.1.1}$$

$$2. \qquad A, B \in \zeta \implies \begin{array}{l} A \cap B \in \zeta \quad \wedge \quad A \cup B \in \zeta \\ \wedge \quad A \backslash B \in \zeta \quad \wedge \quad \mathcal{E} \backslash A \in \zeta \end{array} \tag{3.1.2}$$

$$3. \qquad A_1, A_2 \cdots \in \zeta \implies \bigcap_{\mathtt{i}=1}^{\infty} A_{\mathtt{i}} \in \zeta \quad \wedge \quad \bigcup_{\mathtt{i}=1}^{\infty} A_{\mathtt{i}} \in \zeta \tag{3.1.3}$$

The elements of $\zeta$ are called events in $\zeta$. We define a probability measure as a mapping $\mathcal{P} : \zeta \to [0,1]$ fulfilling the properties for all $A, B \in \zeta$ and $A_1, A_2, \cdots \in \zeta$:

$$1. \qquad \mathcal{P}(\varnothing) = 0 \quad \wedge \quad \mathcal{P}(\mathcal{E}) = 1 \tag{3.1.4}$$

$$2. \qquad A \cap B = \varnothing \implies \mathcal{P}(A \cup B) = \mathcal{P}(A) + \mathcal{P}(B) \tag{3.1.5}$$

$$3. \quad \forall \mathtt{i}, \mathtt{j} : A_{\mathtt{i}} \cap A_{\mathtt{j}} = \varnothing \implies \mathcal{P}\left(\bigcup_{\mathtt{i}=0}^{\infty} A_{\mathtt{i}}\right) = \sum_{\mathtt{i}=1}^{\infty} \mathcal{P}(A_{\mathtt{i}}) \tag{3.1.6}$$

Using this concept we define $(\mathcal{E}, \zeta, \mathcal{P})$ to be a random space.

### 3.1.2 Random Variables

Let $(\mathcal{E}, \zeta, \mathcal{P})$ a random space and $\hat{X} : \mathcal{E} \to \mathbb{R}$ a mapping of elements from $\mathcal{E}$ to a real number. Furthermore let:

$$A : \mathbb{R} \to 2^{\mathcal{E}}; x \mapsto \left\{ e \in \mathcal{E} \;\middle|\; \hat{X}(e) < x \right\} \tag{3.1.7}$$

By this the definition of a **R**andom **V**ariable (RV) is given by:

$$\hat{X} \text{ is a random variable} \quad \Longleftrightarrow \quad \forall x \in \mathbb{R} : A(x) \in \zeta \tag{3.1.8}$$

By using this, a probability distribution function is given by:

$$\Phi : \mathbb{R} \to \mathbb{R}^+, x \mapsto \mathcal{P}(A(x)) \tag{3.1.9}$$

Providing that $\Phi$ is continuous differentiable, the respective **P**robability **D**ensity **F**unction (PDF) is defined as:

$$\varphi := \partial\Phi \tag{3.1.10}$$

In the following, the existence and convergency of the used integrals are assumed. The most important characteristics of random variables are the **mean** $\bar{X}$ and **standard deviation** $\sigma_X$, defined by:

$$\bar{X} := \int_{\mathbb{R}} x \cdot \varphi(x) \cdot dx \tag{3.1.11}$$

$$\begin{aligned} \sigma_X^2 &:= \int_{\mathbb{R}} \left(x - \bar{X}\right)^2 \cdot \varphi(x) \cdot dx \\ &= \overline{\left(\hat{X} - \bar{X}\right)^2} \end{aligned} \tag{3.1.12}$$

The former is also called the expectation of $X$ and the squared standard deviation $\sigma_X^2$ the **variance** of $\hat{X}$

These definitions can be directly generalized to $\mathtt{n}$-dimensional random vectors $\hat{X} = (\hat{X}_1, ..., \hat{X}_\mathtt{n})$, where each $\hat{X}_\mathtt{i}$ is a random variable. For random vectors it holds:

$$A : \mathbb{R}^\mathtt{n} \to 2^\mathcal{E}; x \mapsto \left\{ e \in \mathcal{E} \;\middle|\; \forall \mathtt{i} \in \{1..\mathtt{n}\} : \hat{X}_\mathtt{i}(e) < x_\mathtt{i} \right\} \tag{3.1.13}$$

The corresponding probability distribution function is:

$$\Phi : \mathbb{R}^\mathtt{n} \to \mathbb{R}^+, x \mapsto \mathcal{P}(A(x)) \tag{3.1.14}$$

If a continuous function $\varphi : \mathbb{R}^{\mathtt{n}} \to \mathbb{R}^{+}$ exist, such that

$$\int_{\mathbb{R}^{\mathtt{n}}} \varphi(x)\, dx = 1 \quad \text{and} \quad \forall x \in \mathbb{R}^{\mathtt{n}} \; : \; \Phi(x) = \int_{-\infty}^{x} \varphi(x)\, dx \tag{3.1.15}$$

it is defined to be the PDF of $\Phi$. The density function for a subset $M \subset \{1...\mathtt{n}\}$ of components of $\hat{X}$ results in:

$$\varphi_{X_M}(x_M) = \int_{\mathbb{R}} \cdots \int_{\mathbb{R}} \varphi(x) \cdot \prod_{M'} dx_i \tag{3.1.16}$$

Where $M' := \{1...\mathtt{n}\}/M$ is the inverted set to $M$.

Using this, the mean of $\hat{X}$ can be computed component wise by:

$$\bar{X}_{\mathtt{i}} = \int_{\mathbb{R}} x_{\mathtt{i}} \cdot \varphi_{x_{\mathtt{i}}}(x_{\mathtt{i}}) \cdot dx_{\mathtt{i}} \tag{3.1.17}$$

$$\overset{(3.1.16)}{=} \int_{\mathbb{R}} x_{\mathtt{i}} \cdot \int_{\mathbb{R}} \cdots \int_{\mathbb{R}} \varphi(x) \cdot \prod_{\mathtt{j} \neq \mathtt{i}} dx_{\mathtt{j}} \cdot dx_{\mathtt{i}} \tag{3.1.18}$$

$$= \int_{\mathbb{R}^{\mathtt{n}}} x_{\mathtt{i}} \cdot \varphi(x) \cdot dx \tag{3.1.19}$$

$$\Rightarrow \bar{X} = \int_{\mathbb{R}^{\mathtt{n}}} x \cdot \varphi(x) \cdot dx \tag{3.1.20}$$

Given a linear mapping $A \in \mathbb{R}^{\mathtt{m} \times \mathtt{n}}$ and $a \in \mathbb{R}^{\mathtt{m}}$ the mean of $A \cdot \hat{X} + b$ results in:

$$\overline{A \cdot \hat{X} + b} = \int_{\mathbb{R}^{\mathtt{n}}} Ax + b \cdot \varphi(x) \cdot dx \tag{3.1.21}$$

$$= A \cdot \int_{\mathbb{R}^{\mathtt{n}}} \varphi(x) \cdot dx + b \cdot \int_{\mathbb{R}^{\mathtt{n}}} \varphi(x) \cdot dx \tag{3.1.22}$$

$$= A \cdot \bar{X} + b \tag{3.1.23}$$

For random vectors the **covariance** $\sigma_{X_\mathtt{i}X_\mathtt{j}}$ can be defined as a generalization of the variance:

$$\begin{aligned}\sigma_{X_\mathtt{i}X_\mathtt{j}} &:= \int_{\mathbb{R}^{|\{\mathtt{i},\mathtt{j}\}|}} \left(x_\mathtt{i} - \bar{X}_\mathtt{i}\right)\cdot\left(x_\mathtt{j} - \bar{X}_\mathtt{j}\right)\cdot\varphi_{X_{\{\mathtt{i},\mathtt{j}\}}}\cdot dx_\mathtt{i} dx_\mathtt{j} \qquad (3.1.24)\\ &= \overline{\left(x_\mathtt{i} - \bar{X}_\mathtt{i}\right)\cdot\left(x_\mathtt{j} - \bar{X}_\mathtt{j}\right)}\end{aligned}$$

As can be seen, $\sigma_{X_\mathtt{i}X_\mathtt{i}} = \sigma^2_{X_\mathtt{i}}$ is the variance of component $X_\mathtt{i}$. Using this definition the **correlation** $r_{\mathtt{ij}}$ for two components of $X$ can be defined to be:

$$r_{\mathtt{ij}} := \frac{\sigma_{X_\mathtt{i}X_\mathtt{j}}}{\sigma_{X_\mathtt{i}}\cdot\sigma_{X_\mathtt{j}}} \qquad (3.1.25)$$

For the correlation $r_{\mathtt{ij}} \in [-1,1]$ holds. $\hat{X}_\mathtt{i}$ and $\hat{X}_\mathtt{j}$ are said to be fully correlated if $r_{\mathtt{ij}} \in \{-1,1\}$ and stochastically independent if $r_{\mathtt{ij}} = 0$, and thus $\sigma_{X_\mathtt{i}X_\mathtt{j}} = 0$. Finally the covariance $C$ and correlation $R$ matrices are defined as:

$$C_X := \left(\sigma_{X_\mathtt{i}X_\mathtt{j}}\right)_{\mathtt{ij}} \quad \text{and} \quad R_X := \left(r_{\mathtt{ij}}\right)_{\mathtt{ij}} \qquad (3.1.26)$$

Using (3.1.24) the covariance matrix can be written as:

$$C_X = \overline{\left(\hat{X} - \bar{X}\right)\cdot\left(\hat{X} - \bar{X}\right)^T} \qquad (3.1.27)$$

These matrices fulfill the requirements of positive semi-definiteness (see section 2.1). This is the basic property, that allows for the definition of Gaußian distributions (see section 3.1.3) and the analysis tools discussed in section 3.4.

According to (3.1.23) the covariance $C_Y$ for the transformed RV $Y = A\cdot X + b$ can be determined. This is done by propagating $C_X$ using $A$:

$$\begin{aligned}C_Y &= \overline{\left(A\cdot\hat{X} + b - \overline{A\cdot\hat{X} + b}\right)\cdot\left(A\cdot\hat{X} + b - \overline{A\cdot\hat{X} + b}\right)^T} \qquad (3.1.28)\\ &= A\cdot\overline{\left(\hat{X} - \bar{X}\right)\cdot\left(\hat{X} - \bar{X}\right)^T}\cdot A^T \qquad (3.1.29)\\ &= A\cdot C_X\cdot A^T \qquad (3.1.30)\end{aligned}$$

This covariance propagation is also referred to as linear error propagation.

### 3.1.3 Gaußian Distribution

The Gaußian or normal distribution $\mathcal{N}$ is special, in the sense, that it can be defined using its mean and covariance solely. A detailed description of it's properties and further analysis can be found in [Ton90]. For a Gaußian distributed, $\mathtt{n}$-dimensional random vector $\hat{X}$ the PDF is given by:

$$\varphi(x) = \frac{1}{\sqrt{(2\pi)^{\mathtt{n}} \cdot |C_X|}} \cdot e^{-\frac{1}{2}\left(x-\bar{X}\right)^T C_X^{-1}\left(x-\bar{X}\right)} \tag{3.1.31}$$

Thanks to the fact that a normal distributed variable is characterized by its mean and covariance, these can be used as definition for such a variable:

$$\hat{X} \sim \mathcal{N}\left(\bar{X}, C_X\right) \tag{3.1.32}$$

Because transformations on random variables accordingly apply to their mean and covariance, normal distributions can be transformed in the same way. Given $A \in \mathbb{R}^{\mathtt{m}\times\mathtt{n}}$ and $a \in \mathbb{R}^{\mathtt{m}}$ the mean and covariance of $\hat{Y} = A \cdot \hat{X} + b$ are determined by (3.1.23) and (3.1.30). Thus:

$$\hat{Y} \sim \mathcal{N}\left(A \cdot \bar{X} + b, A \cdot C_X \cdot A^T\right) \tag{3.1.33}$$

As is obvious from equation (3.1.31) the normal distribution is only defined for positive definite (i. e., non singular) covariance matrices $C_X$. I. e., for covariances, that include correlations $r_{\mathtt{ij}} = \pm 1, \mathtt{i} \neq \mathtt{j}$, this definition is invalid. Anyway, a generalization can be defined, using a matrix decomposition (e. g.,**S**ingular **V**alue **D**ecomposition, see [MFW04], [Koc99]). Using the decomposition, the RV and its covariance can be transformed to a lower dimensional random space, where the respective covariance is not degenerated. Therefore, equation (3.1.31) has to be modified, by using the pseudo inverse of $C_X$ and its minimal determinant (product of singular values $\neq 0$).

Note that the resulting probability distribution is defined in the reduced random space. Thus, analysis (e. g., see section 3.4.1) of the respective RV is only possible in its reduced form. By this it is not straight forward to

apply the evaluation results to the original RV.

### 3.1.4 Random Processes

**R**andom **P**rocesses (RP) describe time depending processes driven by random variables. They are often used, to describe physical processes, whose exact properties are unknown. For example the movement of a molecule in a gas or the so called Brownian motion, the movement of Brownian particles, can be modeled this way. By determining characteristics of such trajectories (periodicity, asymptotic behavior etc.) knowledge about its future development can be gained.

In the following, only time discrete processes are discussed. They are used as basis for the Kalman filter theory (see section 3.3). Using a special probability space the probability of undertaking a certain trajectory in $\mathbb{R}^{\mathtt{n}}$ is described. Let $\hat{X}_{\mathtt{k}} \sim \mathcal{N}\left(\bar{X}_{\mathtt{k}}, C_{X_{\mathtt{k}}}\right)$, $\mathtt{k} \in \mathbb{N}_0$ and $\hat{\omega}_{\mathtt{k}} \sim \mathcal{N}\left(0_{\mathtt{n}}, C_{\omega_{\mathtt{k}}}\right)$ series of random variables in $\mathbb{R}^{\mathtt{n}}$. Let $x_0$ an initial value, that is $\mathcal{P}\left(\hat{X}_0 = x_0\right) = 1$, and $F_k \in \mathbb{R}^{\mathtt{n}\times\mathtt{n}}$ transition matrices fulfilling

$$\hat{X}_{\mathtt{k}} \;=\; F_{\mathtt{k}} \cdot \hat{X}_{\mathtt{k}-1} + \hat{\omega}_{\mathtt{k}} \tag{3.1.34}$$

By this the sequences $\hat{\wp} = \hat{X}_0, \hat{X}_1, ...$ and $F_1, F_2, ...$ describe a time discrete random process. Equations (3.1.23) and (3.1.30) imply:

$$\bar{X}_{\mathtt{k}} \;=\; F_{\mathtt{k}} \cdot \bar{X}_{\mathtt{k}-1} + \bar{\omega}_{\mathtt{k}} \;=\; F_{\mathtt{k}} \cdot \bar{X}_{\mathtt{k}-1} \tag{3.1.35}$$

$$C_{X_{\mathtt{k}}} \;=\; F_{\mathtt{k}} \cdot C_{X_{\mathtt{k}-1}} \cdot F_{\mathtt{k}}^T + C_{\omega_{\mathtt{k}}} \tag{3.1.36}$$

It is worth noticing, that the random process is correlated over time. I. e., the correlation

$$C_{X_{\mathtt{k}}, X_{\mathtt{k}+1}} \;=\; C_{X_{\mathtt{k}}} \cdot F^T$$

and by this all $C_{X_{\mathtt{i}}, X_{\mathtt{j}}}$ are not necessarily equal to 0.

The most simple non constant process results from $x_0 = 0_{\mathtt{n}}$, $F_{\mathtt{k}} := 1^{\mathtt{n}}$ and is called random walk (see [Çin10]). For such processes $\bar{X}_{\mathtt{k}} = 0$ holds for all $\mathtt{k}$. They can be used to describe errors evolving from drift effects, whose local changes are independent in time. Such drifts can arise from integration of noisy data, as done in strap down navigation (see 2.4.2,

page 30). Since $C_{X_0} = 0^{\mathtt{n}}$ is used as initial distribution, the covariance for $\hat{X}_{\mathtt{k}}$ is $C_{X_{\mathtt{k}}} = \sum_{\mathtt{i}=1}^{\mathtt{k}} C_{\omega_{\mathtt{k}}}$. Thus, the magnitude of $C_{\omega_{\mathtt{k}}}$ describes the magnitude of drift in $\hat{X}_{\mathtt{k}}$.

For a single dimensioned RP having a constant $C_\omega = \sigma^2$ a standard deviation of $\sqrt{\mathtt{k}\cdot\sigma^2} = \sqrt{\mathtt{k}}\cdot\sigma$ results for $\hat{X}_{\mathtt{k}}$. Relating steps $\mathtt{k} \to \mathtt{k}+1$ with a timeslice of $\delta\tau = 1[s]$, a measure of drift independent of time can be defined. That is, the standard deviation can be normalized by $\mathtt{k}$ seconds. This yields a drift over time of:

$$R_{\mathtt{k}} = \frac{\sqrt{\mathtt{k}}\cdot\sigma}{\mathtt{k}} = \frac{\sigma}{\sqrt{\mathtt{k}}}\left[\frac{[\sigma]}{\sqrt{s}}\right] \tag{3.1.37}$$

Think of a drift process in a one dimensional position. That is, the RP defines a movement, having standard deviation $\sigma[m]$ for increments per second. By this $R$ results in $R = \sigma\left[\frac{m}{\sqrt{s}}\right]$. Thus, after $\tau$ seconds, the RPs standard deviation will be $\sqrt{\tau}\cdot R = \sqrt{\tau}\cdot\sigma[m]$. Using different time slices for $\mathtt{k} \to \mathtt{k}+1$, $R$ is scaled accordingly.

## 3.2 Weighted Least Squares

The so called "Weighted Least Squares" estimators (WLSE) aim at solving imperfect linear equation systems. In conjunction with the simplified Least Squares Estimator, they have a long history in science. In [Sor70] Sorensen discusses the evolution "from its inception by Gauß to its modern form, developed by Kalman" (i. e., the Kalman Filter, see the following section). The description of WLSEs presented here, is taken from [MFW04].

For a given matrix $A \in \mathbb{R}^{\mathtt{n}\times\mathtt{m}}$ and an observation vector $l \in \mathbb{R}^{\mathtt{n}}$ a solution $p \in \mathbb{R}^{\mathtt{m}}$ has to be determined, such that:

$$A\cdot p = l \tag{3.2.1}$$

For a unique solution $p$ the matrix $A$ is required to be of full column rank, i. e., $rank\,(A) = \mathtt{m}$. That is, the equation system (3.2.1) has to consist of a least $\mathtt{n}$ independent equations and thus $\mathtt{n} \geqslant \mathtt{m}$. Fulfilling these requirements

$p$ can be determined by:

$$A^T A \cdot p = A^T \cdot l \quad \Longrightarrow \quad p = \left(A^T A\right)^{-1} A^T \cdot l \tag{3.2.2}$$

The invertibility of $A^T A$ follows from the fact that $A$ has full column rank. Due to erroneous data (e. g., measurement inaccuracy, quantization errors etc.) $p$ can only be determined approximately. Providing $rank\,(A) = \mathtt{m}$, what can be assured by increasing the number $\mathtt{n}$ of equations, (3.2.2) provides the solution with the smallest quadratic error. That is:

$$p = \underset{\pi \in \mathbb{R}^{\mathtt{n}}}{argmin}\ (|A \cdot \pi - l|_2) \tag{3.2.3}$$

Here the vector $z = A \cdot \pi - l$ is often referred to as the vector of residuals or the contradiction for $\pi$.

Moreover, such equation systems are mostly build up using noisy data. To model uncertain observations, $l$ is assumed to be a sample drawn from a random variable $\hat{l} \sim \mathcal{N}(\bar{l}, C_l)$. By this the most likely (given $l$) solution $p$ is a sample drawn from $\hat{p} \sim \mathcal{N}(\bar{p}, C_p)$. It can be determined by:

$$\begin{aligned} A \cdot p = l \quad &\Longrightarrow \quad A^T C_l^{-1} A \cdot p = A^T C_l^{-1} \cdot l \\ &\Longrightarrow \quad p = \left(A^T C_l^{-1} A\right)^{-1} A^T C_l^{-1} \cdot l \end{aligned} \tag{3.2.4}$$

Applying linear error propagation results in $C_p = \left(A^T C_l^{-1} A\right)^{-1}$.

Even though such estimators are designed for linear equation systems, they can be applied to non linear systems by linearization (1. order Taylor expansion). This allows for solving systems as

$$f\,(p) = l \quad \text{and} \quad g\,(p, l) = 0 \tag{3.2.5}$$

for arbitrary indefinitely differentiable functions $f$ and $g$. The former model is called an explicite equation, the latter is known as implicit constraint. Beside the parameter estimation, WLSEs are able to determine the most likely error in the given observations $l$. The derivation of models and a detailed description of the estimators discussed here can be found in [MFW04], [Koc99].

## 3.3 Kalman Filters

Since its introduction in 1960 by R.E. Kalman (see [Kal60]) the **K**alman **F**ilter (**KF**) has proven, to be a powerful tool for estimating stochastic processes. Because the original KF was only applicable to linear processes, more advanced versions, being able to cope with non linearities, have been developed. The most straight forward generalizations of the standard KF are the extended and the iterated extended Kalman Filter, EKF and IEKF. They are discussed detailed in [WB06] and [LBD04]. Their derivations and notations are adopted in the following sections, to give a short introduction to Kalman Filtering.

After these filters have been established, more sophisticated linearization methods have been discussed. Among others, these are the Unscented or Sigma Point KFs (see [JU97], [JJU04], [Mer04]). Such approaches try to overcome the drawbacks for non linear models by using estimators more robust in such cases. Thus, the non linearities are not eliminated, but more complex algorithms are used. This results in increased computational costs and less applicability to real-time demands. In [LBD04] the authors compare the performance of these types of filters and found, that for special conditions the iterated filters perform best. Because these conditions are met for the proposed SLAM systems and real-time performance is mandatory, the Sigma Point KFs are not discussed further. In addition to the pure Kalman Filter theory, in [Ein12] Einecke provides a detailed overview of filter and smoothing theory in general.

### Problem Statement

The $\mathtt{n}$-dimensional stochastic process to be estimated, is represented by a sequence of random variables $\hat{\wp} = \hat{p}_0, \hat{p}_1, ...$ being normal distributed. That is:

$$\hat{p}_\mathtt{k} \sim \mathcal{N}\left(\bar{p}_\mathtt{k}, C^{p_\mathtt{k}}\right) \quad \text{and} \quad \hat{\mu}_\mathtt{k} := \hat{p}_\mathtt{k} - \bar{p}_\mathtt{k} \sim \mathcal{N}\left(0, C_{p_\mathtt{k}}\right) \tag{3.3.1}$$

Moreover, a state transition $F_\mathtt{k}$ and a process noise model $\hat{\omega}_\mathtt{k}$ are postulated:

$$\hat{p}_\mathtt{k} \quad = \quad F_\mathtt{k} \cdot \hat{p}_{\mathtt{k}-1} + \hat{\omega}_\mathtt{k} \qquad \text{with} \qquad \hat{\omega}_\mathtt{k} \sim \mathcal{N}\left(0, C_{\omega_\mathtt{k}}\right) \tag{3.3.2}$$

Due to the monotonic increasing $\mathtt{k}$ this is often referred to as prediction for time index $\mathtt{k}$. By this the sequences $\hat{p}_\mathtt{k}$, $F_\mathtt{k}$ and $\hat{\omega}_\mathtt{k}$ define a stochastic process (see section 3.1.4, page 43). Since the estimation can determine the state sequence only approximately, this estimation is denoted as a sample $\wp$ drawn from $\hat{\wp}$. After predicting $p_\mathtt{k}^-$ using $F_\mathtt{k}$ and $p_{\mathtt{k}-1}$, observations $l_\mathtt{k}$ on this state are used, to estimate the updated filter state $p_\mathtt{k}$. The observations $l_\mathtt{k}$ are assumed, to be sequence of samples from a sequence of normally distributed random variables $\hat{l}_\mathtt{k}$:

$$\hat{l}_\mathtt{k} \sim \mathcal{N}\left(\bar{l}_\mathtt{k}, C^{l_\mathtt{k}}\right) \quad \text{and} \quad \hat{\nu}_\mathtt{k} := \hat{l}_\mathtt{k} - \bar{l}_\mathtt{k} \sim \mathcal{N}\left(0, C_{l_\mathtt{k}}\right)$$

For the parameter update a prediction $l_\mathtt{k}^- = H_\mathtt{k} \cdot p_\mathtt{k}^-$ for the expected observations is computed using the observation model $H_\mathtt{k}$. Afterwards a contradiction $z_\mathtt{k}$ and the according covariance $C_{z_\mathtt{k}}$ is determined as the difference between the predicted and measured observation. This is then used, to estimate the error $p_\mathtt{k}^- - \bar{p}_k$, where the contradiction $z_\mathtt{k}$ is used in the same way as the residuals in WLSE.

### 3.3.1 Standard KF

As stated, the standard KF requires all system models to be linear. Thus, the prediction models $F_\mathtt{k}$ and $H_\mathtt{k}$ for time transfer and observation respectively are represented using matrices. In addition a process noise model $\hat{\omega}_\mathtt{k}, W_\mathtt{k}$ for the errors in prediction is assumed. $\hat{\omega}_\mathtt{k}$ is modeled using zero mean normal distributed noise.

$$\hat{\omega}_\mathtt{k} \sim \mathcal{N}\left(0, C_{\omega_\mathtt{k}}\right)$$

It is used to generalize the process noise (see equation (3.3.2)) by $\hat{\omega}_\mathtt{k} \rightarrow W_\mathtt{k} \cdot \hat{\omega}_\mathtt{k}$. By this the prediction model is defined as:

$$\begin{aligned}
\bar{p}_\mathtt{k} &= F_\mathtt{k} \cdot \bar{p}_{\mathtt{k}-1} + W_\mathtt{k} \cdot \bar{\omega}_\mathtt{k} = F_\mathtt{k} \cdot \bar{p}_{\mathtt{k}-1} && (3.3.3)\\
\hat{p}_\mathtt{k}^- &= F_\mathtt{k} \cdot \hat{p}_{\mathtt{k}-1} + W_\mathtt{k} \cdot \hat{\omega}_\mathtt{k} \\
&= F_\mathtt{k} \cdot \bar{p}_{\mathtt{k}-1} + F_\mathtt{k} \cdot \hat{\mu}_\mathtt{k} + W_\mathtt{k} \cdot \hat{\omega}_\mathtt{k} \\
&= \bar{p}_\mathtt{k} + F_\mathtt{k} \cdot \hat{\mu}_\mathtt{k} + W_\mathtt{k} \cdot \hat{\omega}_\mathtt{k} && (3.3.4)\\
\hat{\mu}_\mathtt{k}^- &:= F_\mathtt{k} \cdot \hat{\mu}_\mathtt{k} + W_\mathtt{k} \cdot \hat{\omega}_\mathtt{k} && (3.3.5)\\
C_{p_\mathtt{k}^-} &:= F_\mathtt{k} \cdot C_{p_{\mathtt{k}-1}} \cdot F_\mathtt{k}^T + W_\mathtt{k} \cdot C_{\omega_\mathtt{k}} \cdot W_\mathtt{k}^T && (3.3.6)
\end{aligned}$$

$$\Rightarrow \quad \hat{p}_{\mathtt{k}}^{-} \sim \mathcal{N}\left(\bar{p}_{\mathtt{k}}, C_{p_{\mathtt{k}}^{-}}\right) \quad \wedge \quad \hat{\mu}_{\mathtt{k}}^{-} \sim \mathcal{N}\left(0, C_{p_{\mathtt{k}}^{-}}\right) \tag{3.3.7}$$

Equation (3.3.3) describes the true process evolution. That is, the means $\bar{p}_{\mathtt{k}}$ and $\bar{\omega}_{\mathtt{k}} = 0$ are used. According to that, the observation model results in:

$$\begin{aligned}
\bar{l}_{\mathtt{k}} &= H_{\mathtt{k}} \cdot \bar{p}_{\mathtt{k}} && (3.3.8)\\
\hat{l}_{\mathtt{k}} &= \bar{l}_{\mathtt{k}} + \hat{\nu}_{\mathtt{k}} \\
&= H_{\mathtt{k}} \cdot \bar{p}_{k} + \hat{\nu}_{\mathtt{k}} \\
&\overset{(3.3.4),(3.3.5)}{=} H_{\mathtt{k}} \cdot \left(\hat{p}_{\mathtt{k}}^{-} - \hat{\mu}_{\mathtt{k}}^{-}\right) + \hat{\nu}_{\mathtt{k}} \\
&= H_{\mathtt{k}} \cdot \hat{p}_{\mathtt{k}}^{-} - H_{\mathtt{k}} \cdot \hat{\mu}_{\mathtt{k}}^{-} + \hat{\nu}_{\mathtt{k}} && (3.3.9)
\end{aligned}$$

For the filter update the contradiction $\hat{z}_{\mathtt{k}}$ is computed using:

$$\begin{aligned}
\hat{z}_{\mathtt{k}} &:= \hat{l}_{\mathtt{k}} - H_{\mathtt{k}} \cdot \hat{p}_{\mathtt{k}}^{-} && (3.3.10)\\
&= \bar{l}_{\mathtt{k}} + \hat{\nu}_{\mathtt{k}} - H_{\mathtt{k}} \cdot \bar{p}_{\mathtt{k}}^{-} - H_{\mathtt{k}} \cdot \hat{\mu}_{\mathtt{k}}^{-} && (3.3.11)\\
&= \hat{\nu}_{\mathtt{k}} - H_{\mathtt{k}} \cdot \hat{\mu}_{\mathtt{k}}^{-} \quad \Rightarrow \quad \bar{z}_{\mathtt{k}} = 0 && (3.3.12)
\end{aligned}$$

$$\Longrightarrow \quad \hat{z}_{\mathtt{k}} \sim \quad \mathcal{N}\left(0, H_{\mathtt{k}} \cdot C_{p_{\mathtt{k}}^{-}} \cdot H_{\mathtt{k}}^{T} + C_{y_{\mathtt{k}}}\right) \tag{3.3.13}$$

When applying these equations to the estimated process $\wp$, nothing is known about the random variables $\hat{\omega}_{\mathtt{k}}$ and $\hat{\nu}_{\mathtt{k}}$. Thus their known means $\bar{\omega}_{\mathtt{k}} = 0$ and $\bar{\nu}_{\mathtt{k}} = 0$ are used as best guesses. Finally the process of Kalman filtering (see [Kal60]) is defined to be:

$$\begin{aligned}
p_{\mathtt{k}}^{-} &= F_{\mathtt{k}} \cdot p_{\mathtt{k}-1} && (3.3.14)\\
C_{p_{\mathtt{k}}^{-}} &= F_{\mathtt{k}} \cdot C_{p_{\mathtt{k}-1}} \cdot F_{\mathtt{k}}^{T} + W_{\mathtt{k}} \cdot C_{\omega_{\mathtt{k}}} \cdot W_{\mathtt{k}}^{T} && (3.3.15)\\
z_{\mathtt{k}} &= l_{\mathtt{k}} - H_{\mathtt{k}} \cdot p_{\mathtt{k}}^{-} && (3.3.16)\\
C_{z_{\mathtt{k}}} &= H_{\mathtt{k}} \cdot C_{p_{\mathtt{k}}^{-}} \cdot H_{\mathtt{k}}^{T} + C_{l_{\mathtt{k}}} && (3.3.17)
\end{aligned}$$

Finally the estimation, mostly referred to as Kalman Update, is performed using:

$$K_{\mathtt{k}} = C_{p_{\mathtt{k}}^{-}} \cdot H_{\mathtt{k}}^{T} \cdot \left(H_{\mathtt{k}} \cdot C_{p_{\mathtt{k}}^{-}} \cdot H_{\mathtt{k}}^{T} + C_{l_{\mathtt{k}}}\right)^{-1} \tag{3.3.18}$$

$$
\begin{aligned}
z_{\mathtt{k}} &= l_k - H_{\mathtt{k}} \cdot p_{\mathtt{k}}^{-} && (3.3.19)\\
p_{\mathtt{k}} &= p_{\mathtt{k}}^{-} + K_{\mathtt{k}} \cdot z_{\mathtt{k}} && (3.3.20)\\
C_{p_{\mathtt{k}}} &= (1^{\mathtt{n}} - K_{\mathtt{k}} \cdot H_{\mathtt{k}}) \cdot C_{p_{\mathtt{k}}^{-}} \cdot (1^{\mathtt{n}} - K_{\mathtt{k}} \cdot H_{\mathtt{k}})^T + K_{\mathtt{k}} \cdot C_{l_{\mathtt{k}}} \cdot K_{\mathtt{k}}^T && (3.3.21)\\
&= (1^{\mathtt{n}} - K_{\mathtt{k}} \cdot H_{\mathtt{k}}) \cdot C_{p_{\mathtt{k}}^{-}} && (3.3.22)
\end{aligned}
$$

This estimation method can be deduced from equation (3.2.4) using the model (see [PK10]):

$$
A = \begin{pmatrix} -I \\ H_{\mathtt{k}} \end{pmatrix} \quad l = \begin{pmatrix} p_{\mathtt{k}}^{-} \\ l_{\mathtt{k}} \end{pmatrix} \quad p = p_{\mathtt{k}} \qquad (3.3.23)
$$

Equation (3.3.21), known as Joseph's Form, is often used in filter implementations. It is obtained directly by linear error propagation for equations (3.3.19) and (3.3.20). It is more robust to numerical issues and ensures positive semi definiteness for $C_{p_{\mathtt{k}}}$ at the costs of increased computational effort.

### 3.3.2 Extended Kalman Filters

The **E**xtended **K**alman **F**ilter (EKF) was developed, to allow for close to linear system and observation models. To do so, the models are linearized using their first order Taylor expansion to deliver the model matrices.

The prediction and observation models

$$
\begin{aligned}
\hat{p}_{\mathtt{k}} &= f_{\mathtt{k}}\left(\hat{p}_{\mathtt{k}-1}, \hat{\omega}_{\mathtt{k}}\right) && (3.3.24)\\
\hat{l}_{\mathtt{k}} &= h_{\mathtt{k}}\left(\hat{p}_{\mathtt{k}}, \hat{\nu}_{\mathtt{k}}\right) && (3.3.25)
\end{aligned}
$$

are linearized using the Jacobian matrices

$$
\begin{aligned}
F_{\mathtt{k}} &= \left.\frac{f_{\mathtt{k}}(p,0)}{\partial p}\right|_{p=\hat{p}_{\mathtt{k}-1}} \qquad W_k = \left.\frac{f_{\mathtt{k}}(\hat{p}_{\mathtt{k}-1},\omega)}{\partial \omega}\right|_{\omega=0} && (3.3.26)\\
H_{\mathtt{k}} &= \left.\frac{h_{\mathtt{k}}(p,0)}{\partial p}\right|_{p=\hat{p}_k} \qquad V_{\mathtt{k}} = \left.\frac{h_{\mathtt{k}}(\hat{p}_{\mathtt{k}},\nu)}{\partial \nu}\right|_{\nu=0} && (3.3.27)
\end{aligned}
$$

Using these Jacobians as model matrices and replacing equations (3.3.14)

and (3.3.19) by

$$p_{\mathtt{k}}^{-} = f_{\mathtt{k}}\left(p_{\mathtt{k}-1}, 0\right) \tag{3.3.28}$$

$$z_{\mathtt{k}} = l_{\mathtt{k}} - h_{\mathtt{k}}\left(p_{\mathtt{k}}^{-}, 0\right) \tag{3.3.29}$$

the EKF is applied in the same way as the standard KF. If $V_{\mathtt{k}}$ is not equal to the identity mapping, the observation noise $\hat{\nu}_{\mathtt{k}}$ in the KF model has to be substituted using $V_{\mathtt{k}} \cdot \hat{\nu}_{\mathtt{k}}$. This model can be used, to take evolving linearization errors into account.

### 3.3.3 Iterated Extended Kalman Filters

Because the EKF uses the first order Taylor expansion for linearization, it is still error prone for higher non linear models. The **I**terated **E**xtended **K**alman **F**ilter (IEKF) reduces this effect by solving the update process using iterated linearization. Because this is only possible for the update estimation, the IEKF is still prone to linearization errors in the prediction model.

To apply the update process iteratively, the iteration initialization is done using $p^{(0)} = p_{\mathtt{k}}^{-}$. For the following estimation steps $\imath$ the Jacobians for the respective $p^{\imath}$ are computed:

$$H_{\mathtt{k}}^{(\imath)} = \left.\frac{h_{\mathtt{k}}\left(p, 0\right)}{\partial p}\right|_{p=p^{(\imath)}} \qquad V_{\mathtt{k}}^{(\imath)} = \left.\frac{h_{\mathtt{k}}\left(p^{(\imath)}, \nu\right)}{\partial \nu}\right|_{\nu=0} \tag{3.3.30}$$

The update for iteration $\imath$ is done using:

$$z_{\mathtt{k}}^{(\imath)} = l_{\mathtt{k}} - h_{\mathtt{k}}\left(p^{(\imath)}\right) - H_{\mathtt{k}}^{(\imath)} \cdot \left(p^{(0)} - p^{(\imath)}\right) \tag{3.3.31}$$

$$C_{z_{\mathtt{k}}}^{(\imath)} = C_{l_{\mathtt{k}}} + H_{k}^{(\imath)} \cdot C_{p_{\mathtt{k}}^{-}} \cdot {H_{\mathtt{k}}^{(\imath)}}^{T} + V_{\mathtt{k}}^{(\imath)} \cdot C_{\nu_{\mathtt{k}}} \cdot {V_{\mathtt{k}}^{(\imath)}}^{T} \tag{3.3.32}$$

$$K_{\mathtt{k}}^{(\imath)} = C_{p_{\mathtt{k}}^{-}} \cdot {H_{\mathtt{k}}^{(\imath)}}^{T} \cdot {C_{z_{\mathtt{k}}}^{(\imath)}}^{-1} \tag{3.3.33}$$

$$p^{(\imath+1)} = p^{(0)} + K_{\mathtt{k}}^{(\imath)} \cdot z_{\mathtt{k}}^{(\imath)} \tag{3.3.34}$$

$$\tag{3.3.35}$$

As can be seen, the first iteration $\imath = 0$ simplifies to the EKF update

equations. The iteration is continued, until $p^{(\imath)} - p^{(\imath-1)}$ falls beneath a certain threshold or a maximum number of iterations has been reached. Finally the covariance for the filter state updated in iteration $\imath$ is:

$$C_{p_{\mathtt{k}}} = \left(1^{\mathtt{n}} - K_{\mathtt{k}}^{(\imath)} H_{k}^{(\imath)}\right) C_{p_{\mathtt{k}}^{-}} \tag{3.3.36}$$

$$= \left(1^{\mathtt{n}} - K_{\mathtt{k}}^{(\imath)} H_{\mathtt{k}}^{(\imath)}\right) C_{p_{\mathtt{k}}^{-}} \left(1^{\mathtt{n}} - K_{\mathtt{k}}^{(\imath)} H_{\mathtt{k}}^{(\imath)}\right)^{T} + K_{\mathtt{k}}^{(\imath)} C_{y_{\mathtt{k}}} {K_{\mathtt{k}}^{(\imath)}}^{T} \tag{3.3.37}$$

To test a threshold for $p^{(\imath)} - p^{(\imath-1)}$, a vector norm has to be chosen, that is independent on dimension of $p$. This is necessary, since filters with varying state sizes (i. e., size of environment in SLAM) should have a constant threshold. Due to this, the root of mean squares (RMS) is used throughout this thesis.

### 3.3.4 Orientation Representation

In the proposed Kalman Filters incremental rotations (see section 2.2.1) are used as parameters in probabilistic estimation. Thus, a convention has to be defined, allowing for the representation of the uncertainty of such rotations embedded in the filter state $p$. After introducing this convention, the derivatives for incremental rotations are discussed, since they are needed for linear error propagation in the filtering process.

Let $\Phi = \{R, \phi\}$ an incremental rotation and $\hat{p} = (\,\hat{\pi}\;\hat{\phi}\,)$, where $\hat{\phi}$ and $\hat{\pi}$ are random variables and $\phi$ a sample from $\hat{\phi}$. Let:

$$\hat{\phi} \sim \mathcal{N}\left(\bar{\phi}, C_{\phi}\right) \qquad \hat{p} \sim \mathcal{N}\left(\begin{pmatrix} \bar{\pi} \\ \bar{\phi} \end{pmatrix}, \begin{pmatrix} C_{\pi} & C_{\pi,\phi} \\ C_{\phi,\pi} & C_{\phi} \end{pmatrix}\right)$$

Normalizing $\Phi$ also results in a transformation $\hat{p} \to \hat{p}'$, i. e., $\hat{\phi} \to \hat{\phi}'$. By this the representation of the orientations uncertainty in $C_p$ has to be transformed accordingly. The transformation on $\hat{p}$ is done using:

$$\hat{p}' = \begin{pmatrix} \hat{\pi} \\ \hat{\phi}' \end{pmatrix} = \begin{pmatrix} \hat{\pi} \\ R_{(\phi)}{}^{T} \cdot \left(\hat{\phi} - \phi\right) \end{pmatrix} \tag{3.3.38}$$

$$\Phi \quad = \quad \left\{ R \cdot R_{(\phi)}, R_{(\phi)}{}^T \cdot (\phi - \phi) \right\} \quad = \quad \{ R \cdot R_{(\phi)}, 0 \} \tag{3.3.39}$$

As can be seen, this transformation is linear, having the Jacobian:

$$J \quad = \quad \begin{pmatrix} 1 & 0 \\ 0 & R_{(\phi)}{}^T \end{pmatrix}$$

Applying linear error propagation results in:

$$C_{p'} \quad = \quad J \cdot C_p \cdot J^T \quad = \quad \begin{pmatrix} 1 & 0 \\ 0 & R_{(\phi)}{}^T \end{pmatrix} \cdot C_p \cdot \begin{pmatrix} 1 & 0 \\ 0 & R_{(\phi)}{}^T \end{pmatrix}^T \tag{3.3.40}$$

$$= \quad \begin{pmatrix} 1 & 0 \\ 0 & R_{(\phi)}{}^T \end{pmatrix} \cdot C_p \cdot \begin{pmatrix} 1 & 0 \\ 0 & R_{(\phi)} \end{pmatrix} \tag{3.3.41}$$

Thus, the covariance $C_\Phi$ for the rotation $\Phi$ has to be adopted to the actual representation. In the following, this rotation normalization and the respective covariance transformation are applied each time $\Phi$ is modified. This is needed especially, when predictions and updates are performed in the process of Kalman Filtering.

As stated above, derivatives for $\{R, \phi\}$ have to be used for error propagation. These are given by:

$$\left. \frac{\partial \{R, \phi\}}{\partial \phi} \right|_\phi \quad = \quad R \cdot \left. \frac{\partial R_{(\phi)}}{\partial \phi} \right|_\phi$$

$$= \quad R \cdot \left[ \left. \frac{\partial R(\phi)}{\partial \phi(1)} \right|_{\phi(1)} \;\middle|\; \left. \frac{\partial R(\phi)}{\partial \phi(2)} \right|_{\phi(2)} \;\middle|\; \left. \frac{\partial R(\phi)}{\partial \phi(3)} \right|_{\phi(3)} \right] \tag{3.3.42}$$

$$\left. \frac{\partial \{R, \phi\}^T}{\partial \phi} \right|_\phi \quad = \quad \left. \frac{\partial R^T{}_{(\phi)}}{\partial \phi} \right|_\phi \cdot R^T$$

$$= \quad \left[ \left. \frac{\partial R(\phi)}{\partial \phi(1)} \right|_{\phi(1)} \;\middle|\; \left. \frac{\partial R(\phi)}{\partial \phi(2)} \right|_{\phi(2)} \;\middle|\; \left. \frac{\partial R(\phi)}{\partial \phi(3)} \right|_{\phi(3)} \right]^T \cdot R^T \tag{3.3.43}$$

## 3.4 Analysis Tools

For weighted least squares estimators many tools for analyzing the estimation process have been studied, e. g., see [MFW04], [Koc99], [Cot04]. Their use

ranges from consistency tests to covariance component estimation. As was shown in [PK10], these tools can be applied to the process of Kalman filtering also. They can be used for outlier detection and model noise estimation respectively. Since the assumption of linear system models is crucial for such estimators, methods for analyzing the models linearity are needed. In the following, applications to the KFs are proposed, that can be used to analyze the filtering process.

### 3.4.1 Statistical Testing

Least square estimation is heavily error prone for violations of the assumption of Gaußian distributed noise. In especially a single erroneous measurement can invalidate the complete estimation. This is due to the fact, that a quadratic norm is minimized in equation (3.2.3) increasing the weight of unexpected high errors. Thus, methods are required to detect such violations of model assumptions.

Detection of outliers in observations can be performed by testing their probability distribution for being Gaußian. This is done using the $\chi^2$-test (see [MFW04], [Çin10]). This test exploits the fact, that for a $\mathtt{n}$-dimensional Gaußian random variable $\hat{z} \sim \mathcal{N}(0, C_z)$, its squared specialized vector norm follows a $\chi^2_{\mathtt{n}}$-distribution. That is:

$$|\hat{z}|^2_{C_z} \quad = \quad \hat{z}^T \cdot C_z^{-1} \cdot \hat{z} \;\sim\; \chi^2_{\mathtt{n}} \tag{3.4.1}$$

$$\Longrightarrow \quad \mathcal{P}\left(|\hat{z}|^2_{C_z} \leqslant \epsilon\right) \;=\; \int\limits_{[0,\epsilon]} \chi^2_{\mathtt{n}}(\xi) \cdot d\xi \;=:\; \Xi_{\mathtt{n}}(\epsilon) \tag{3.4.2}$$

For a detailed description of the $\chi^2_{\mathtt{n}}$-distribution see [BS96]. This constraint can be used to validate the consistency of samples $z$ from $\hat{z}$. For a given probability $\mathtt{p}$ a threshold $\epsilon$ for the specialized vector norm can be determined using the inverse mapping $\Xi^{-1}_{\mathtt{n}}$. It holds:

$$\Xi^{-1}_{\mathtt{n}}(\mathtt{p}) \;=\; \epsilon \quad \Longleftrightarrow \quad \mathcal{P}\left(|z|^2_{C_z} \leqslant \epsilon\right) \;=\; \mathtt{p} \tag{3.4.3}$$

$$\Longleftrightarrow \quad \mathcal{P}\left(|z|^2_{C_z} > \epsilon\right) \;=\; 1 - \mathtt{p} \tag{3.4.4}$$

This implies, that $z$, fulfilling $|z|^2_{C_z} \leqslant \epsilon$, corresponds to a sample of $\hat{z}$ at a probability of at least $1 - \mathtt{p}$.

This technique for statistical testing is especially suitable for sequential estimation as Kalman Filtering. This is due to the fact, that the predicted observation $l_{\mathtt{k}}^-$ and the observation $l_{\mathtt{k}}$ (see section 3.3) have the same mean $\bar{l}_{\mathtt{k}}$. Thus, equation (3.4.1) applies to the specialized norm for the contradiction $z_k$ and its covariance $C_{z_k}$. By this outliers in the measurements can be detected with a certain probability by using the observation model $h_{\mathtt{k}}$ and the predicted filter state $p_{\mathtt{k}}^-$.

As discussed in 3.1.2 and 3.1.3 the Gaussian distribution can be generalized to degenerated covariance matrices. For such random variables $C_z^{-1}$ has to be replaced in (3.4.1) by the pseudo inverse. This pseudo norm then follows a $\chi^2_{\mathtt{k}}$ distribution, where $\mathtt{k}$ is the rank of the covariance matrix $C_z$.

### 3.4.2 Covariance Consistency

The consistency of the estimated parameters $p$, being samples from a random variable $\hat{p}$ with the estimated covariance matrix $C_p$, is an important requirement for the used estimator. That is, the error $\hat{z} := \hat{p} - \bar{p}$ has to be normally distributed with covariance $C_p$ for a correct estimation. Due to the violation of model assumptions and numerical issues the estimated covariance $C_p$ will not match the true error covariance $\hat{C}_p$ exactly. Thus, a quality measure for consistency has to be defined. This is done using two different characteristics.

**Consistency** In the following, estimations RV $\hat{p}$ (an estimation is a sample from $\hat{p}$) and an estimated covariance $C_p$ are consistent, if $C_p$ is an upper bound for the true error distribution. That is, $C_p$ is at least as large as the true error covariance $\hat{C}_p$. This is done, using the matrix ordering defined in section 2.1.

$$\hat{p}, C_p \text{ are consistent} \quad \Longleftrightarrow \quad \hat{z} = (\hat{p} - \bar{p}) \sim \mathcal{N}\left(0, \hat{C}_p\right) \ \wedge \ \hat{C}_p \leqslant C_p \tag{3.4.5}$$

This can be checked by testing the eigenvalues of $C_p - \hat{C}_p$ to be positive (see section 2.1). Moreover, this constraint is equivalent to $\forall z : |z|_{\hat{C}_p} \leqslant |z|_{C_p}$

and thus:

$$\forall \mathtt{p}: \quad \mathcal{P}\left(|z|_{\hat{C}_p} \leqslant \Xi^{-1}(\mathtt{p})\right) \geqslant \mathcal{P}\left(|z|_{C_p} \leqslant \Xi^{-1}(\mathtt{p})\right) \tag{3.4.6}$$

By this accepting $z$ as a sample of $\hat{z}$ using $C_p$ implies acceptance using $\hat{C}_p$. That is, $C_p$ does not accept a sample, that is rejected by $\hat{C}_p$. A similar definition and further discussions on covariance consistency can be found in [PB11]. Note that the consistency is a binary decision and allows the estimated variances to be unbounded. Thus, a second characteristic has to be used, to determine the quality of $C_p$.

**Informativity** Let $M$ a set of samples from a random variable $\hat{p}$. Moreover, for a probability $\mathtt{p}$ let

$$N_{\mathtt{p}}^{C_p} := \left\{ p \in M \;\middle|\; |p - \bar{p}|^2_{C_p} \leqslant \Xi^{-1}(\mathtt{p}) \right\} \tag{3.4.7}$$

the set of samples in M accepted by $C_p$ with a the probability of $1 - \mathtt{p}$. By this $N_{\mathtt{p}}^{C_p}$ is expected to contain $100{\cdot}\mathtt{p}$ percent of samples in $M$ (see equation (3.4.3)). The informativity of the sample set $M$ and the given covariance $C_p$ is defined as the vector:

$$\mathcal{I}(M, C_p) := \begin{pmatrix} 100 \cdot \frac{\left|N_{0.3829}^{C_p}\right|}{|M|} - 38.29 \\ 100 \cdot \frac{\left|N_{0.6827}^{C_p}\right|}{|M|} - 68.27 \\ 100 \cdot \frac{\left|N_{0.9545}^{C_p}\right|}{|M|} - 95.45 \\ 100 \cdot \frac{\left|N_{0.9973}^{C_p}\right|}{|M|} - 99.73 \end{pmatrix} \in \begin{pmatrix} [-38.29 \ldots 61.71] \\ [-68.27 \ldots 31.73] \\ [-95.45 \ldots 4.55] \\ [-99.73 \ldots 0.27] \end{pmatrix} \tag{3.4.8}$$

The closer to zero the components of $\mathcal{I}(M, C_p)$ are, the more informative is $C_p$ with respect to $M$. Moreover, a value $\geqslant 0$ implies a consistent and a value $< 0$ an inconsistent covariance assumption with respect to the used probability bound. Note that this informativity measure is given in percent. Thus, it allows for a comparison of informativity for different estimation models delivering covariances in different magnitudes.

Applying this measure to a one dimensional random variable and its

**Table 3.1.** Sigma scales $s$ and the percentage $100\mathcal{P}\left(|p-\bar{p}| \leqslant s \cdot \sigma_p\right)$ of expected inliers.

| $s$ | 0.5 | 1 | 2 | 3 |
|---|---|---|---|---|
| inliers [%] | 38.29 | 68.27 | 95.45 | 99.73 |

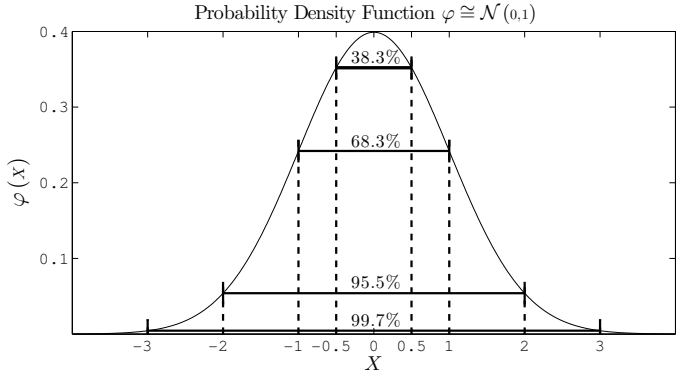


**Figure 3.1.** Visualization for sigma bounds $[\frac{1}{2}\sigma, \sigma, 2\sigma, 3\sigma]$ for a single variate random variable $X$. It is visualized for 0-mean and a standard deviation 1, i. e., $X \sim \mathcal{N}_{(0,1)}$.

standard deviation $\sigma_p$, the set $N_{\mathtt{p}}^{\sigma^2}$ simplifies to:

$$N_{\mathtt{p}}^{\sigma^2} = \left\{ p \in M \;\middle|\; |p-\bar{p}|_{C_p} \leqslant \Xi^{-1}{}_{(\mathtt{p})} \right\} \tag{3.4.9}$$

$$= \left\{ p \in M \;\middle|\; \frac{|p-\bar{p}|^2}{\sigma^2} \leqslant \Xi^{-1}{}_{(\mathtt{p})} \right\} \tag{3.4.10}$$

$$= \left\{ p \in M \;\middle|\; |p-\bar{p}| \leqslant \sqrt{\Xi^{-1}{}_{(\mathtt{p})}} \cdot \sigma \right\} \tag{3.4.11}$$

with the factor $s_{\mathtt{p}} := \sqrt{\Xi^{-1}{}_{(\mathtt{p})}}$. The product $s_{\mathtt{p}} \cdot \sigma_p$ is often called the $s_{\mathtt{p}}$-sigma bound and $\left[-s_{\mathtt{p}} \cdot \sigma_p \ldots s_{\mathtt{p}} \cdot \sigma_p\right]$ the respective $s_{\mathtt{p}}$-confidence interval. The bounds mostly used for evaluation are visualized in table 3.1 and figure 3.1. These bounds also motivate the choices of $\mathtt{p}$ in the definition of

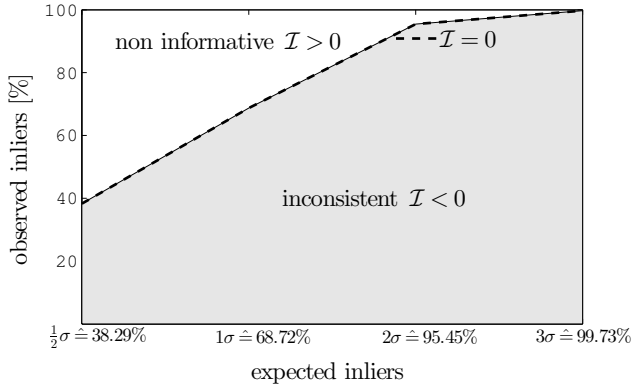


**Figure 3.2.** Visualization of the informativity with respect to the expected percentage of inliers (see equation (3.4.8)). For a single dimensioned random variable, the inlier expectation corresponds to the $\sigma$-bounds.

the informativity measure (equation (3.4.8)). For a visualization of the relationship between the inlier percentage, the $\sigma$-bounds and the informativity $\mathcal{I}$ see figure 3.2.

**Consistency vs Informativity** Using both characteristics, the quality of the estimated covariance can be evaluated. The informativity measure provides partial information on the consistency and the informativity itself for certain designated probabilities (or sigma bounds). That is, a notion of distance between the assumed and true system covariance is given. For its evaluation no knowledge of the true covariance is needed, but only the predicted one. Thus, it is especially useful for analysis with unknown ground truth statistics.

The consistency decision on the other hand provides information on the overall validity of the covariance as an upper bound for errors. It is not limited to the chosen points of evaluation (choices of `p` for informativity). Its drawback is the need for determining the true estimations covariance. For some cases it is possible to determine it empirically using large sample

sets (numerical integration of equation (3.1.27)). In other cases, i. e., real world experiments, it is not possible to generate sample clouds large enough for determining stable empirical covariances.

**NEES** Another consistency measure is the so called NEES (**N**ormalized **E**stimation **E**rror **S**quared, e. g., see [Bai+06], [Sol+12]). Instead of computing an inlier ratio based on the $\chi^2$ test, they use the squared specialized vector norm $\epsilon_k := |\hat{z_k}|^2_{C_{z_k}}$, where $z_k$ is the estimation error for test run $k$. After performing several test runs, they determine an average NEES $\bar{\epsilon}$. Moreover, they define an interval $I$, such that $\bar{\epsilon} \in I$ marks a consistent and informative estimation. This measure provides only an abstract valuation of consistency and informativity and is not comparable to other tests, since it is dependent on the size of $z$. Moreover, the interpretation of the result is a binary decision, i. e., $\bar{\epsilon} \in I$ or $\bar{\epsilon} \notin I$. Due to this, the combination of consistency checks and the informativity measure $\mathcal{I}$ is preferred to the NEES.

#### Covariance Consistency Methods

Least squares estimators are heavily error prone to invalid models and inconsistent covariances. In especially correlations, that are not considered in the models or used covariances, result in inconsistent estimations. Such correlations can arise from biased observations or model approximations, e. g., assuming slowly varying entities as constants in sequential estimation (KF).

When the sources of such inconsistencies are known, covariance consistency methods (e. g., Covariance Intersection, see [Uhl03],[PB11]) can be applied. They guarantee consistent estimation even in the presence of unknown correlations. A drawback of these methods is, that the estimated covariances are inflated, resulting in a loss of informativity. This effect can be reduced, by using the Partitioned Covariance Intersection (see [PB11]), providing that knowledge on the structure of correlations is available. Two other methods are known as State Augmentation (see below) and Measurement Differencing (see [Wen07]) respectively. The former introduces nuisance parameters to model colored (i. e., time correlated) noise, and can be thought of estimating the biases. This method can be applied at low

costs, but requires the biases to be observable. The latter uses differences of observations instead the observations directly, to remove the bias. This reduces the problem of colored noise but also the information introduced to the update, e. g., position increments are used instead of absolute positions.

**State Augmentation** is a frequently used method, to circumvent system errors, that arise from correlated noise. Such correlations are mostly introduced by biased prediction models or observations. Providing that a process model for the biases evolution is available, e. g., constant bias in the most simple case, it can be incorporated in the filter state. That is, the filter attempts to estimate the bias and possibly its process parameters. This method has two major drawbacks. On the one hand, additional parameters cause increased computational effort. On the other hand, non observable augmentation parameters can cause the filter to diverge. Nevertheless, for observable biases and with acceptable computational effort, its a adequate technique to meet the uncorrelated noise requirement. A more detailed description can be found in [GA01].

### 3.4.3 Linearization Analysis

Because the assumption of linear system models is substantial for least squares estimation, the quality of linearization has to be determined for non linear systems. Therefore, the basic model for linearization in WLSE is used (see [MFW04], [Koc99]):

$$f(p) \;=\; l \quad \leftarrow \text{linearization} \rightarrow \quad A \cdot p = l \quad ; \quad A := \left.\frac{\partial f}{\partial}\right|_{p} \tag{3.4.12}$$

Using two instances $(p_0, l_0)$ and $(p_1, l_1)$ of this model results in:

$$\begin{aligned} f(p_0) = l_0\,,\; f(p_1) = l_1 \quad &\longleftrightarrow \quad A \cdot p_0 = l_0\,,\; A \cdot p_1 = l_1 \\ &\Downarrow \\ f(p_1) - f(p_0) = l_1 - l_0 \quad &\longleftrightarrow \quad A \cdot (p_1 - p_0) = l_1 - l_0 \end{aligned} \tag{3.4.13}$$

Using $\Delta p := p_1 - p_0$, this can be transformed to:

$$f(p_0 + \Delta p) - f_{(p_0)} \quad \stackrel{!}{=} \quad A \cdot \Delta p \tag{3.4.14}$$

Computing $f(p_0 + \Delta p) - f(p_0) - A \cdot \Delta p$ results in the remainder of the first order Taylor expansion for $f$ and $\Delta p$ in $p_0$ (see [BS96]). Since linearization is based on the assumption, that this remainder is negligible, it is trivial measure of linearizion quality.

Linearized error propagation of $\hat{p}$, $C_p$ (see equation (3.1.30)) assumes $f$ to be linear at $p_0 = \bar{p}$. Moreover, $\Delta p$ has to be large, compared to the uncertainty region of $C_p$, i. e., $|\Delta p|_{C_p} \to \infty$. By defining

$$\mathcal{LE}_f(p_0, \Delta p) \;:=\; 100 \cdot \begin{pmatrix} \frac{A(1,:)\Delta p - f_1(p_0+\Delta p) + f_1(p_0)}{f_1(p_0+\Delta p) - f_1(p_0)} \\ \vdots \\ \frac{A(\mathtt{N},:)\Delta p - f_{\mathtt{N}}(p_0+\Delta p) + f_{\mathtt{N}}(p_0)}{f_{\mathtt{N}}(p_0+\Delta p) - f_{\mathtt{N}}(p_0)} \end{pmatrix} \tag{3.4.15}$$

a relative measure in percent for the quality of linear error propagation at $p_0$ with the expected error $\Delta p$ results for each component of $f$. Note that for $f_{\mathtt{N}}(p_0 + \Delta p) - f_{\mathtt{N}}(p_0) \longrightarrow 0$ the quality might be undefined, i. e., for constant mappings. Thus, such singularities have to be taken into account, when analyzing the linearity of a function. Since an analytical limiting process is needed, to evaluate $\mathcal{LE}$ in such nodes, they will be excluded from evaluation when using numerical methods.

In [CDM08] and [CDM07] a similar linearization measure was used. Instead of using the full Taylor expansion remainder $f_1(p_0 + \Delta p) - f_1(p_0)$, the fraction between the first and second order derivative is evaluated. This requires the computation of the second order derivative for $f$. Moreover, they determine the derivative only for a single parameter, instead of using the full directed derivative. This prohibits the use of the covariance to determine $\Delta p$ (correlations between $\Delta p_{(\mathtt{i})}$), since the one dimensional derivative only takes the respective standard deviation into account. Solà et al. proposed a linearization measure in [Sol+12], that exploits the actual multivariate covariance matrix. Nevertheless, it is restricted to the analysis of the filter's observation models, i. e., the contradiction $z$ (see equation (3.3.29), page 50). Due to this, $\mathcal{LE}$ will be used in the following linearization analysis.

As can be seen, for a non constant and linear mapping $f$ and an arbitrary $\Delta p$ follows $\mathcal{LE}_f(p_0, \Delta p) = 0$. Moreover, values greater than 0 indicate, that the propagated absolute error $|A \cdot \Delta p|$ exceeds the true absolute error $|f(p_0 + \Delta p) - f(p_0)|$ and has the correct sign (direction). By this the propagated error becomes non informative but provides a valid bound. A value $< 0$ indicates, that the predicted error is too small or even wrongly signed and does not provide a valid error bound. Thus, the measure $\mathcal{LE}$ for linearity corresponds to the informativity $\mathcal{I}$ for consistency analysis.

Chapter 4

# System Design

When designing systems for SLAM, several considerations have to be taken into account. First of all, a decision on the specialization for a specific environment has to be made. When adjusting the design to a restricted application area, information on the specific environment can be incorporated. By this the performance of the system can be improved at the price of losing generality. This tradeoff has impact on every part of the system to be developed. This includes hardware and models as well as software.

In this chapter the models and methods used for the proposed SLAM system are introduced. At first the general hardware setup for an IMU-aided visual SLAM system is discussed. Following that, the mathematical models for state representation and estimation methods with respect to the hardware setup are introduced. Finally, the error models, that are used in the designed Kalman Filter, are discussed.

## 4.1 Hardware

As already stated, the used hardware setup is made of a stereo camera system and a low cost IMU, based on the MEMS technology (see section 2.4.2). Compared to monocular visual SLAM, this setup provides several advantages. Though using two cameras and an IMU increases the costs, the benefits outweigh this additional effort. Moreover, since digital cameras and IMUs are applied to various areas (cell phones, game controlers ...) and increasing quantity, their cost decreased significantly during the past years. Thus, especially for tasks, where high precision and robustness are required, inertial aided stereo SLAM systems outperform plain monocular approaches.

### 4.1.1 Stereo Camera Systems

In the following sections an ideal stereo system is assumed. In especially, the stereo correspondences delivered by feature tracking techniques are given in normalized pixels. This can be achieved as discussed in section 2.4.1, page 26. Moreover, the coordinate transformation $\Gamma_c^{c'}$ between the master camera $c$ and the slave camera $c'$ is given by a 1-dimensional translation (baseline $b$):

$$\Gamma_c^{c'} \quad = \quad \left\langle \Phi_c^{c'}, t_{cc'}^c \right\rangle \quad = \quad \left\langle 1^3, \begin{pmatrix} b \\ 0 \\ 0 \end{pmatrix} \right\rangle$$

Although real stereo camera systems do not match this assumption exactly, the resulting deviations can be undone as described in section 2.4.1 on page 27. Due to the simplicity of pixel normalization and stereo alignment for single features, the rectification process can be applied online. In the following the advantages and drawbacks of stereo camera systems are discussed. Afterwards the design decision for a stereo vision system is founded.

**Advantages** Stereo cameras outmatch monocular vision in various areas. The most important one is the direct observation of depth for stereo correspondences, allowing for immediately creation of 3D features, i. e., landmarks (see section 6.7.2, page 142). Moreover, the determined 3-space entities can be computed in metrical scale, providing that the stereo systems baseline is known. By this the stereo measurements can be easily combined with other sensors, that deliver data in metric scale, as the IMUs used for this thesis. This is a major advantage compared to monocular scene reconstruction, since it overcomes the so called scale ambiguity. As discussed by Hartley and Zisserman in [HZ03], this ambiguity cannot be solved without additional information. Moreover, for sequential reconstruction of 3D scenes, as the environment maps in SLAM, the scale factor undergoes a drift, which incorporates unrecoverable errors.

Due to the usage of two images per time step, the visual redundancy can be exploited. When visual features are tracked in images over time, erroneous matching can occur. For monocular vision such mismatches can only be detected, by using predictions of the cameras movement. For stereo

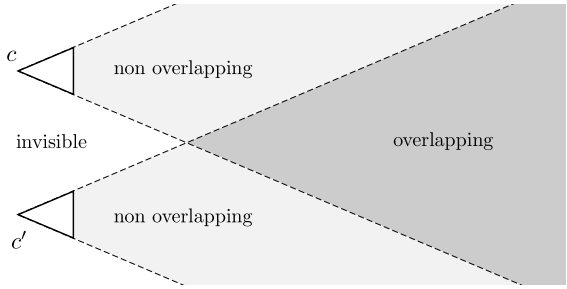


**Figure 4.1.** Example for non overlapping views in stereo cameras. As can be seen, decreasing the camera distance, results in and increased overlap of fields of view.

vision the stereo constraints for epipolar geometry and disparity (see 2.3.3) can be exploited. E. g., the positions of stereo correspondences in the images height in normalized pixels are known to be the same (at least similar for imperfect measurements).

Finally, recorded stereo image sequences can be used, to create complete 3D models of the environment with high accuracy. Although the camera's estimated trajectories can be used for 'pseudo' stereo reconstructions (e. g., see [HZ03], [Sze11]), well calibrated stereo rigs deliver reconstructions of much higher accuracy. Modern methods for real-time dense stereo (e. g., see [Bud12]) can be used, to create camera centric 3D models online. By this additional information for SLAM algorithms can be gathered.

**Drawbacks** The disadvantages of stereo vision are non overlapping camera views, limitations due to the systems baseline, and increased computational effort. Non overlapping views result from the fact, that cameras at distinct positions and the same orientation cannot have the same field of vision (see figure 4.1). By this parts of the acquired images cannot be used for stereo vision. Although this effect can be reduced by using smaller baselines, this ends up in a further drawback. The stereo systems accuracy in depth measurements mainly depends on the image resolution and the baseline. Though the image resolution is fixed by the used cameras, the baseline can be increased, to improve the depth resolution.

A small baseline reduces the effect of non overlapping views, but also decreases the depth resolution. This is due to the fact, that smaller baselines result in smaller disparities and that the ability, to resolve disparities, is limited by the image resolution. Thus, small baselines are suitable for scenes with small point to camera distances, allowing the exploitation of a maximal image overlap. Vice versa, large baselines decrease the view overlapping but increase depth perception. Thus, they are suitable for huge scenes, having large point to camera distances. Keeping in mind these antagonisms, the baseline has to be chosen as a tradeoff between depth resolution, overlapping fields of view and spatial limitations.

Finally, the computational effort for stereo image processing is increased compared to monocular vision. To deliver the additional information using two images, most of the image processing has to be applied twice. I. e., for feature tracking, a stereo match has to be performed in addition. Another task with high computational effort is image rectification and undistortion (see section 2.4.1) of the whole image. Although these can be done in a single step, it has to be applied for both images. Nevertheless, for the proposed system only the detected and matched image features are rectified. By this the computational effort can be reduced, such that it's negligible compared to feature detection.

**Design Decision** The most important impact on the design decision was given by the stereo visions ability to directly observe metric depths. By this the landmarks to be used, can be initialized, once they have been detected. This can be done without any assumptions on an expected depth and the usage of very uncertain depth initializations, as is discussed in (see [CDM08], [Sol+12]). Moreover, the metric scale of the SLAM state space, defined by the landmarks, allows for the usage of IMU measurements. Even though methods have been proposed, to use IMUs in scale ambiguous environments (see [Mar11], [Nüt+11]), they suffer from restrictive assumptions. Such restrictions are hardly fulfilled in the real world and lead to inconsistent system models.

The drawback of limited depth perception, depending on the used baseline and image resolution, is accepted for initialization of landmarks. Observing landmarks over time results in additional depth information in the same way as for monocular systems. Thus, this limitation would not be cir-

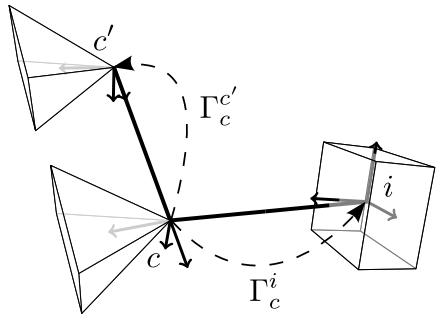


**Figure 4.2.** Stereo camera setup with master $c$ and slave $c'$ camera and inertial measurement unit $i$.

cumvented by monocular systems, but the direct initialization of landmarks would be of reduced quality. The improved ability to detect outliers due the redundancy in stereo vision is another important factor. This increases the stability and robustness of the SLAM system especially in environments with ill-conditioned visual appearance (e. g., see section 7.2, page 172).

Finally, the stereo constraints allow for the specialized landmarks parametrization proposed in section 4.2.1. It is adopted to the properties of a stereo camera system, by exploiting these constraints for improving the involved models applicability to linear estimators.

### 4.1.2 IMU Stereo System

The hardware setup used for real-world experiments is schematically visualized in figure 4.2. The master camera $c$ is rigidly coupled with the slave camera $c'$. The external calibration of stereo properties and the internal camera parameters have been determined using the methods discussed in section 2.4.1. As stated before, in the following it is assumed that the respective calibrations have been used, to rectify all feature points used for landmark matching. The inertial measurement unit $i$ is rigidly coupled with the stereo rig. The calibration of the IMU to camera coordinate transformation has been done, using the software proposed in section 6.9, page 148. The stereo calibration can be used, to rectify the used features in advance

to applying the linear estimator. Nevertheless, this does not apply to the IMU to camera calibration. Due to its comparably low precision, it has to be incorporated in the system models for error modeling as discussed in 4.2.6 (page 80) and A.2.2 (page 219).

Compared to the decision for stereo vision the advantages of IMUs for Kalman Filtering are more obvious. Once the decision for stereo vision has been made, the IMU can be incorporated directly to the filter process. By this the quality and consistency of the systems movement prediction can be improved significantly. This is due to the fact, that movement models without IMUs have to make certain assumptions. The most popular movement model, called dead reckoning, was adopted from the early vessel navigators. It makes use of the assumption of constant velocity for spatial and angle domain respectively. This is obviously an invalid assumption for freely moving SLAM systems and has to be compensated by a high process noise for velocities. By this the consistency of the filter can hardly be held. Moreover, when visual features are not available on sub tracks, the constant velocity model will result in poor or even invalid predictions. Using an IMU the consistency and prediction quality can be kept even for long time segments (see chapter 7).

The only limiting factor when dealing with IMUs is the additional effort in costs. Since IMUs are available in many price classes, a device with the appropriate precision and costs can be chosen. Robot platforms used for SLAM are mostly more cost-intensive than the stereo cameras and a low cost IMU. Thus, for such systems the comparably small amount of increased costs take a back seat.

## 4.2 Modeling

After specifying the used hardware platform, the mathematical models, needed for state estimation via the Kalman Filter, can be deduced. At first the models of state parametrizations are given. That is, the different landmark and navigation representations are discussed. Following that, the prediction and observation models ($f$ and $h$ respectively in chapter 3.3) are introduced. Finally the methods for relocalization of the local navigation frame $n$ and the error models are defined. The coordinate frames used for

modeling are those, defined in section 2.2. To improve the readability in the following definitions, the transpose operator is omitted for stacked vectors. Moreover, the indices for the discrete time steps $k$ are omitted, when these are clear from the context.

### 4.2.1 Landmarks

A fundamental part of SLAM is building up an environment map during the navigation process. For non visual systems various representations for objects are used. In visual SLAM these are mostly limited to so called landmarks, denoted as $\mathcal{X}$ in the following. They represent positions $X$ in Euclidean 3-space, which can be associated with features in the environment. For visual SLAM these features are chosen in a way, that allows for tracking them, using image processing. For landmarks two basic parametrizations were established. They are discussed in the following. Subsequent a new landmark representation, that has been designed especially for stereo SLAM, is proposed.

#### ES Landmarks

The most intuitive representation in **E**uclidean **S**pace (**ES**) emerged with the first SLAM systems and has been used a long time (see [Dis+01], [Sch+07]). It makes use of the landmarks position in the global coordinate system $e$ given as Euclidean vector. Thus, the landmark parametrization for the filter state is:

$$\mathcal{X}^{\mathrm{ES}} \quad = \quad t^e_{eX} \tag{4.2.1}$$

Even though it is straight forward to implement and uses well known mathematical models, its use in linear stochastic estimation is suboptimal (see [HMR08], [Bai+06], [CDM08]). This is due to the fact, that noise resulting from triangulation of noisy stereo correspondences can hardly be represented using zero mean Gaussian noise.

#### ID Landmarks

The **I**nverse **D**epth (**ID**) parametrization, developed in 2006 by Montiel et al. (see [MCD06], further analysis in [CDM08]), is designed to overcome this

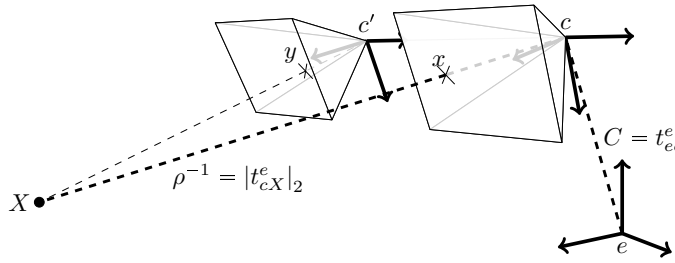


**Figure 4.3.** Visualization of landmark parametrization for ID. $x \leftrightarrow y$ stereo correspondence, $c$ and $c'$ actual camera frames. The hookup point $\mathcal{C}$ and inverse depth $\rho$. View ray angles $\phi, \psi$ parametrize direction of $t^e_{cX}$.

major drawback. From then on, ID has become the de facto standard for visual SLAM. Various systems where proposed for different application areas (e. g., see [Art+09], [LGK11], [SLP07]).

Montiel et al. defined ID landmarks, to be represented by a position in 3-space, a view ray and the inverse depth. They choose the position $\mathcal{C} := t^e_{ec}$ of the camera, the landmark was firstly observed from, as "anchor". The view ray, parametrized by angles $\phi$ and $\psi$, describes the direction pointing from $\mathcal{C}$ to $X$, i. e., the spherical coordinates of $t^e_{cX} \cdot |t^e_{cX}|_2^{-1}$. The inverse depth $\rho = |t^e_{cX}|_2^{-1}$ describes the inverse distance between the camera and the landmark. That is, the ID representation of $X$ is:

$$\mathcal{X}^{\text{ID}} = \begin{pmatrix} \mathcal{C} & \phi & \psi & \rho \end{pmatrix} \tag{4.2.2}$$

$$= \begin{pmatrix} t^e_{ec} & sphere\left(|t^e_{cX}|_2^{-1} \cdot t^e_{cX}\right) & |t^e_{cX}|_2{}^{-1} \end{pmatrix} \tag{4.2.3}$$

See figure 4.3 for a visualization of the involved entities. Even though this parametrization matches the unbiased noise assumption much better compared to ES, it suffers from more non linear models and over parameterization. That is, 6 parameters are used for the representation of 3 DOF. This can end up in violations of assumptions made by the estimators (like KF). Moreover, the computational effort is strongly increased. A further drawback is the possible inconsistency of estimated anchors. That is, two

landmarks initialized with the same anchor from the same pose will end up with different anchors.

#### Other Global Parametrizations

Even though further representations have been discussed, none of them was able to resume IDs predominance. In [Mar+08] and [Mar+09] Marzorati et al. discuss the Inverse Scaling parametrization. It makes use of homogeneous points in $\mathbb{P}^3$ as landmark representatives. As for ID, this uses more parameters for estimation than DOFs for the solution exist. Moreover, it is scale invariant resulting in estimation ambiguities when no additional constraints are applied.

In [IBN09] Imre et al. propose two improved ID parametrizations. To reduce the drawback of IDs over parametrization, they use a common anchor for landmarks initialized from the same camera. In a more "strict" definition, they also include the cameras orientation as rotation anchor to the "super-landmark". Both approaches suffer from an increased effort for managing landmarks within the state, i. e., semantics of anchors and rays etc.. Nevertheless, inconsistent anchor estimations (see ID) are circumvented.

In [Cer+11] Ceriani et al. discuss different point and line parametrizations, that are used in monocular SLAM. Since line features are out of the scope of this thesis, they won't be discussed further. In addition to the ES, ID and homogeneous points they discussed the "Anchored Homogeneous Points". These are an intermediate level between homogeneous points and ID. By this they suffer from the same ambiguities and over parametrization as ID.

All these parametrizations are defined in the global reference system. Moreover, ES is the traditional landmark representation and ID became the standard parametrization in SLAM. Due to this, only ES and ID will be analyzed further in this thesis.

#### PD Landmarks

To overcome the drawbacks of ES and ID, a new parametrization, that has been specialized for stereo SLAM, is proposed in the following. It is called **P**oint-**D**isparity space, **PD** for short. A 3-space point in the cameras coordinate system $c$ is defined by its projection $x^{\mathrm{PD}}$ to the master camera $c$

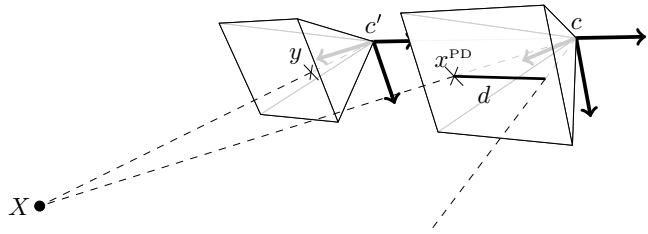


**Figure 4.4.** Visualization of entities involved in landmark parametrization for PD. $x = x^{\mathrm{PD}} \leftrightarrow y = x^{\mathrm{PD}} + \left(\begin{smallmatrix} d \\ 0 \end{smallmatrix}\right)$ stereo correspondence, $c$ and $c'$ actual camera frames.

(2D image point) and its disparity $d$ (see section 2.3.3) to the image of the slave camera $c'$. Thus, for stereo correspondences $x \leftrightarrow y$ we define:

$$\mathcal{X}^{\mathrm{PD}} = \begin{pmatrix} x^{\mathrm{PD}} \\ d \end{pmatrix} = \begin{pmatrix} x \\ x_{(1)} - y_{(1)} \end{pmatrix} \tag{4.2.4}$$

Similar to ID this representation combines the camera pose, the view direction to the landmark and its depth using disparity. In contrast to ID, PD uses the actual pose as reference for the landmarks eliminating the need for an anchor point. The involved entities are visualized in figure 4.4. This way the landmark is represented using 3 DOF, the minimal parametrization for 3-space points. Moreover, since the disparity has not to be computed using Euclidean distances, it is determined by the image measurements in a linear fashion. By this $x^{\mathrm{PD}}$ and $d$ can be extracted from image data directly, i. e., without non linear mappings, and PD landmarks can be related to vision measurements more easily.

**Inverse Depth vs Disparity** As mentioned above, ID and PD model the landmark to camera distance in a similar way. On the one hand, ID uses the inverse of the vector norm of the camera to landmark translation. That is, it encodes the view rays inverse length. On the other hand, PD models the disparity, which is proportional to the inverse of the minimal distance between the camera plane and the landmark.

By this the meaning of both representations are metrical different. As

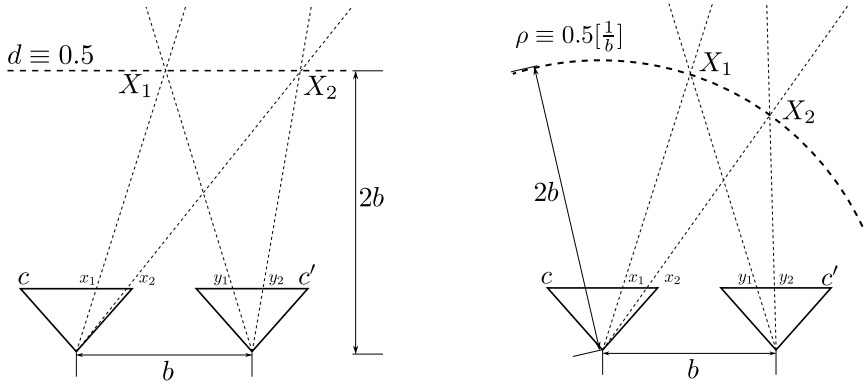


**Figure 4.5.** Surfaces of constant disparity $d \equiv 0.5$ compared to those for constant inverse depth $\rho \equiv 0.5$. The iso-disparity defines a plane $\left\{ X \in \mathbb{R}^3 \;\middle|\; X_{(3)} = \frac{b}{d} \right\}$ relative to the cameras $c$ and $c'$. The iso-inverse-depth defines a sphere of radius $\rho^{-1}$, centered at the position of $c$.

stated above, the disparity directly results from the subtractions of the first components of the stereo correspondences. In contrast to that, the distance to the landmark has to be computed explicitly for ID. This is especially apparent, when visualizing the iso-disparity and iso-inverse-depth surfaces (see figure 4.5). The constant disparity $d$ defines planes in 3-space parallel to the camera plane, having a distance $\frac{b}{d}$ for baseline $b$. By this the initialization of $d$ is independent on the view rays direction. In contrast to that, a constant inverse depth $\rho$ defines 3-space spheres, having the radius $\rho^{-1}$. By this the initialization of the inverse depth is dependent on the distance between the stereo correspondences (the disparity) and their direction. See section 4.2.4 for a comparison of both initialization models.

**Points at Infinity or Behind the Camera** The ID and PD landmark models allow for inverse depths and disparities being $\leqslant 0$. Even though this is a valid state in projective geometry, it is difficult to transfer to the real world. The singularity at 0 marks the points at infinity, discussed in section 2.3.2, page 19. These points define a view direction, but no analogon in Euclidean space and thus the real world exists. Nevertheless, the projective imaging is

still valid.

On the one hand, such points at infinity can be used to model landmarks, being too distant, to resolve the difference between the stereo correspondences $x$ and $y$. For such far points, the resulting disparity falls beneath the image resolution. Nevertheless, as discussed in literature (see [MD06], [Tul+08]), they provide rich information on the observing systems orientation.

On the other hand, $\rho, d < 0$ result in the points behind the camera. Such points are valid in Euclidean space, but do not fit the projective space model. This is due to the fact, that homogeneous points are scale invariant, thus:

$$\begin{pmatrix} x(1) \\ x(2) \\ 1 \end{pmatrix} = -1 \cdot \begin{pmatrix} x(1) \\ x(2) \\ 1 \end{pmatrix} = \begin{pmatrix} -x(1) \\ -x(2) \\ -1 \end{pmatrix}$$

This results in points in front of and behind the camera being mapped to the same image point. Thus, the imaging process of negative depths and disparities, cannot be modeled by projection matrices.

From these inconsistencies in the projective geometry a problem for linear estimation arises. Due to the uncertainty of the estimated entities, the uncertainty area can contain negative inverse depths or disparities. Thus, it cannot be ensured, that the estimation is valid, i. e., $\rho, d \geqslant 0$. This has to be taken into account, when estimating landmarks.

#### Other Local Parametrizations

Local parametrizations of map structures and landmarks have been studied in literature. In [Mei+11], Mei et al. propose a stereo SLAM algorithm, that uses a relative bundle adjustment and a graph based representation of local maps. They use 3-space landmarks in the local camera frame to represent landmarks, seen in the respective camera. Since a main focus of this thesis is the analysis of consistency for KF based SLAM, the comparison to bundle adjustment techniques is improper. These methods do not provide valid covariance information, such that they can hardly be included in the consistency analysis.

In [CNT04] and [Cas+07] Castellanos et. al. propose an algorithm, that makes use of local sub maps, that are given relative to certain system

poses. This way a set of small maps arises, that are later joined to a global map. They showed, that the consistency of KFs for SLAM can be improved by using local maps. They used laser scanner sensors for an evaluation of their methods and do not discuss the influence of landmark parametrization. Thus, their results cannot be related to the analysis, performed in this thesis.

### 4.2.2 Navigation State

Beside the $\mathtt{N}$ landmarks $\mathcal{X}_\mathtt{i}$ the most important parameters to be estimated determine the systems pose. The pose is represented by Euclidean vectors for the positions and incremental rotations (see section 2.2, page 11) for the orientation. In addition to the pose entities like velocity and nuissance parameters have to be incorporated in the filter state. They are used to provide the pose predictions and help to minimize estimation errors this way.

Throughout this thesis two different navigation models will be used. On the one hand, a global navigation state $\pi^g$ is used for state representations of **ES** and **ID** models. It parameterizes the global position, velocity and orientation. Keep in mind, that the global navigation frame $g$ and the navigation frame $n$ are identical for global navigation models (see 2.2.2). On the other hand, a local navigation state $\pi^l$, needed for the local **PD** landmarks, is used. For local navigation the navigation frame $n$ is used as reference and differs from the global frame $e$. Beside the global position and orientation, a velocity and pose increments are used. The latter describe the system movement, that occurred after the last relocalization (see chapter 4.2.7).

**The global parametrizations** define the pose by the IMUs coordinate transformation $\Gamma_i^e$. That is, the translation $t_{ei}^e$ and the incremental rotation $\Phi_i^e$ determine the filter states position and orientation. In addition to this, the global velocity $v_{ei}^e$ is included in the navigation state. Thus, the navigation information is made up of position, velocity and orientation. By this the global navigation parametrization ends up with:

$$\pi^g = \begin{pmatrix} t_{ei}^e & v_{ei}^e & \Phi_i^e \end{pmatrix} \tag{4.2.5}$$

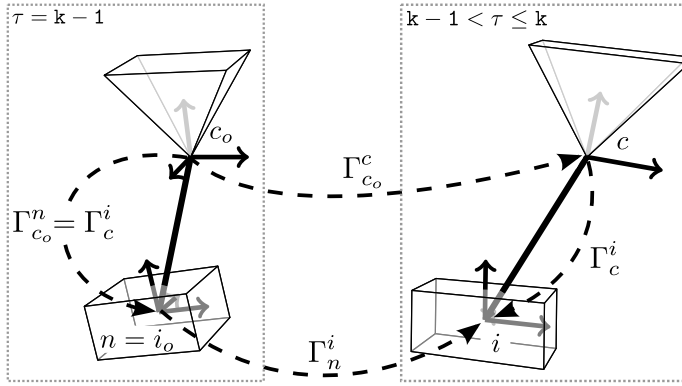


**Figure 4.6.** Definition of navigation frame $n$ for local parametrization using the IMU frame $i_o$ at time of update $\mathtt{k}-1$ and the respective camera frame $c_o$. Moreover, the actual IMU $i$ and camera $c$ frame for time up to $\mathtt{k}^{\mathrm{th}}$ update.

**The local parametrization** introduces an intermediate navigation coordinate frame $n$ (see chapter 2.2.2). The $n$-frame corresponds to the IMU-frame $i_o$ at the point in time of the previous estimation step (see figure 4.6). The global pose is given by the navigation frames global coordinate transform $\Gamma^e_n$. The incremental pose is determined by the coordinate transform $\Gamma^n_i$ for the IMU frame $i$ with respect to $n$. Using the time invariant IMU to camera calibration $\Gamma^c_i$, the incremental pose can be used, to relate the landmarks given locally in the old camera frame $c_o$ to the new camera frame $c$. This is done as described for the observation models in section 4.2.6. By this the local navigation parametrization ends up with:

$$\pi^l = \begin{pmatrix} t^n_{ni} & v^n_{ni} & \Phi^n_i & t^e_{en} & \Phi^e_n \end{pmatrix} \tag{4.2.6}$$

### 4.2.3 Filter State

To improve the filters quality, modeling parameters are introduced to the filter state in addition to the landmarks and the navigation state.

The IMU nuisance parameters, as introduced in section 2.4.2, are used,

to compensate for systematical errors in IMU measurements. These are especially the biases $b_a$ and $b_w$ for the accelerations and turn rates respectively. Additionally the scale factors $s_a$ and $s_w$ can be included in the estimation. Depending on the type and quality of the used IMU, these factors are close to 1 such that they have impact only on long term navigation. In the following,

$$\eta \;=\; \begin{pmatrix} b_a & b_w & s_a & s_w \end{pmatrix} \tag{4.2.7}$$

will be used, to denote the IMUs nuisance parameters.

To reduce the impact of systematical errors in the observation models, the IMU-camera calibration (see 6.9) is integrated to the filter state. Therefore, $t_{ci}^c$ and $\Phi_i^c$ are included in the parameters to be estimated. By this the uncertainties of the determined calibration and the evolving correlations over time are incorporated into the error models (see section 4.3).

Combining the introduced estimation parameters the final filter state $p$ is given by the $\mathtt{i} \in \{1..\mathtt{N}\}$ landmarks $\mathcal{X}_\mathtt{i}$, the navigation state $\pi$, the nuisance parametrization $\eta$ and the IMU-camera calibration $\Gamma_i^c$:

$$p \;=\; \begin{pmatrix} \pi & \eta & \Phi_i^c & t_{ci}^c & \mathcal{X}_1 & \dots & \mathcal{X}_\mathtt{N} \end{pmatrix} \tag{4.2.8}$$

### 4.2.4 Landmark Initialization

When new landmarks are detected, these have to be incorporated in the filter state. Therefore, stereo correspondences $x \leftrightarrow y$ and the actual filter state are used, to initialize the respective landmark in ES-, ID- and PD-space respectively. As for the introduction of the observation models (section 4.2.6) a simplified model is given. The full initialization including the IMU-camera calibration $\Gamma_i^c$ is discussed in appendix A.2.1.

**For ES** representations the landmarks 3-space position has to be triangulated explicitly. Thanks to the stereo constraints this can be accomplished using:

$$\mathcal{X}^{\text{ES}} \;=\; \frac{b}{x(1) - y(1)} \cdot \Phi_c^e \cdot \begin{pmatrix} x(1) \\ \frac{1}{2} \cdot (x(2) + y(2)) \\ 1 \end{pmatrix} + t_{ec}^e \tag{4.2.9}$$

The respective Jacobian used for initializing the landmarks covariance is given in A.3.1 on page 222.

**For ID** representations the actual system position $t^e_{ec}$ is used for $\mathcal{C}$ (see page 70). Additionally the view direction $\phi$,$\psi$ and the inverse depth $\rho$ have to be computed. The view direction is given by:

$$\phi = \arctan\left(\mathtt{x}_{(1)}, \mathtt{x}_{(3)}\right) \tag{4.2.10}$$

$$\psi = \arctan\left(-\mathtt{x}_{(2)}, \sqrt{\mathtt{x}_{(1)}{}^2 + \mathtt{x}_{(3)}{}^2}\right) \tag{4.2.11}$$

using the two argument tangens inverse and

$$\mathtt{x} = \Phi^e_c \cdot \begin{pmatrix} x_{(1)} \\ \frac{1}{2}\cdot\left(x_{(2)} + y_{(2)}\right) \\ 1 \end{pmatrix} \tag{4.2.12}$$

The inverse depth is determined by:

$$\frac{1}{\rho} = \left| \frac{b}{x_{(1)} - y_{(1)}} \cdot \Phi^e_c \cdot \begin{pmatrix} x_{(1)} \\ \frac{1}{2}\cdot\left(x_{(2)} + y_{(2)}\right) \\ 1 \end{pmatrix} \right|_2 \tag{4.2.13}$$

$$= \frac{b}{x_{(1)} - y_{(1)}} \cdot \left| \begin{pmatrix} x_{(1)} \\ \frac{1}{2}\cdot\left(x_{(2)} + y_{(2)}\right) \\ 1 \end{pmatrix} \right|_2 \tag{4.2.14}$$

$$= \frac{b}{x_{(1)} - y_{(1)}} \cdot \sqrt{x_{(1)}{}^2 + \frac{1}{4}\cdot\left(x_{(2)} + y_{(2)}\right)^2 + 1} \tag{4.2.15}$$

$$\Rightarrow \rho = \frac{x_{(1)} - y_{(1)}}{b \cdot \sqrt{x_{(1)}{}^2 + \frac{1}{4}\cdot\left(x_{(2)} + y_{(2)}\right)^2 + 1}} \tag{4.2.16}$$

Using these identities, the inverse depth landmark is determined by:

$$\mathcal{X}^{\text{ID}} = \begin{pmatrix} t_{ec}^{e} \\ \arctan_2\left(\mathtt{x}_{(1)}, \mathtt{x}_{(3)}\right) \\ \arctan_2\left(-\mathtt{x}_{(2)}, \sqrt{\mathtt{x}_{(1)}{}^2 + \mathtt{x}_{(3)}{}^2}\right) \\ \frac{x(1)-y(1)}{b\sqrt{x(1)^2+\frac{1}{4}(x(2)+y(2))^2+1}} \end{pmatrix} \tag{4.2.17}$$

The respective Jacobian matrix is given in appendix A.3.1 on page 223.

**For PD** representations the initialization can be done using the stereo correspondence $x \leftrightarrow y$ solely:

$$\mathcal{X}^{\text{PD}} = \begin{pmatrix} x^{\text{PD}}(1) \\ x^{\text{PD}}(2) \\ d \end{pmatrix} = \begin{pmatrix} x(1) \\ \frac{1}{2} \cdot \left(x(2) + y(2)\right) \\ x(1) - y(1) \end{pmatrix} \tag{4.2.18}$$

As can be seen, in contrast to ES and PD initialization this initialization model is strictly linear. Moreover, the Jacobian for this initialization model vanishes for the filter state. Thus, the sub matrix for the stereo correspondences is given by:

$$J_{xy}^{\text{PD}} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 & \frac{1}{2} \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{4.2.19}$$

### 4.2.5 Prediction

A prediction of the navigation state can improve the estimations quality. Since linearization methods are used for parameter estimation, a precise prediction influences its convergence speed and quality. Moreover, for large state changes the prediction can be outside the convergence region. That is, an invalid least squares estimation is determined, possibly causing filter divergency.

A movement prediction aided by IMU measurements allows for precise estimation. Here the strap down navigation model described in 2.4.2 is used. For simplicity time indexing and unchanged parameters are omitted.

Moreover, all state components, except for the navigation state, are left unchanged in the prediction model $f$. Thus, only the respective sub model $f^{\pi}$ is given.

For the **global navigation** state keep in mind, that the global frame $e$ and the navigation frame $n$ are identical. Using equations (2.4.7),(2.4.12) and (2.4.15) the pose altered by a single prediction step of length $\delta\tau$ is determined by:

$$f^{\pi}\left(\pi^{g},\eta\right)=\begin{pmatrix} t^{e}_{ei}+\delta\tau\cdot v^{e}_{ei}+\frac{1}{2}\delta\tau^{2}\cdot\left(\Phi^{e}_{i}\left(s_{a}a+b_{a}\right)-\gamma^{e}\right)\\ v^{e}_{ei}+\delta\tau\cdot\left(\Phi^{e}_{i}\left(s_{a}a+b_{a}\right)-\gamma^{e}\right)\\ \Phi^{e}_{i}+\delta\tau\cdot\left(s_{w}w+b_{w}\right)\end{pmatrix} \tag{4.2.20}$$

The **local navigation** state predicts the incremental pose $\Gamma^{n}_{i}=\langle\Phi^{n}_{i},t^{n}_{ni}\rangle$ and the velocity $v^{n}_{ni}$ solely. This way the coordinate transformation between two camera measurements can be determined. Similar to equation (4.2.20), the local prediction is given by:

$$f^{\pi}\left(\pi^{l},\eta\right)=\begin{pmatrix} t^{n}_{ni}+\delta\tau\cdot v^{n}_{ni}+\frac{1}{2}\delta\tau^{2}\cdot\left(\Phi^{n}_{i}\left(s_{a}a+b_{a}\right)-{\Phi^{e}_{n}}^{T}\gamma^{e}\right)\\ v^{n}_{ni}+\delta\tau\cdot\left(\Phi^{n}_{i}\left(s_{a}a+b_{a}\right)-{\Phi^{e}_{n}}^{T}\gamma^{e}\right)\\ \Phi^{n}_{i}+\delta\tau\cdot\left(s_{w}w+b_{w}\right)\end{pmatrix} \tag{4.2.21}$$

Note that in contrast to the global navigation the gravity vector $\gamma^{e}$ is transferred to the navigation frame using ${\Phi^{nT}_{e}}$. Moreover, in the local prediction $\Gamma^{e}_{n}=\langle\Phi^{e}_{n},t^{e}_{en}\rangle$ is left unchanged. Thus, the navigation frame has to be relocalized at certain points in time. This is typically done when image measurements have been acquired, see chapter 4.2.7 (page 86) for details.

### 4.2.6 Observation

As visual SLAM systems observe their environment using cameras, the measurements used for state estimation are given by matches of the landmarks $\mathcal{X}$ to the acquired images. For stereo vision a stereo correspondence $x\leftrightarrow y$ is matched. The observation models $h$ predict the matches positions in the images by using the systems navigation state prediction $\pi$. From this the

observation constraint

$$x, y \quad \text{match} \quad \mathcal{X} \implies \begin{pmatrix} x \\ y \end{pmatrix} \stackrel{!}{=} h(\pi, \mathcal{X}) \tag{4.2.22}$$

for each matched landmark results.

The full observation model $h(p)$ observes all landmarks $\mathcal{X}_\mathtt{i}$ using the predicted pose. It can be determined by stacking all $h(\pi, \mathcal{X}_\mathtt{i})$, that is:

$$h(p) = \begin{pmatrix} h(\pi, \mathcal{X}_1) \\ \vdots \\ h(\pi, \mathcal{X}_\mathtt{N}) \end{pmatrix} \stackrel{!}{=} \begin{pmatrix} x_1 \\ y_1 \\ \vdots \\ x_\mathtt{N} \\ y_\mathtt{N} \end{pmatrix} \tag{4.2.23}$$

This model is used for the Kalman Update (see sections 3.3.2 and 3.3.3). Therefore, the assumed observation noise $\nu$ in equation (3.3.25) is applied additive. Thus, the Kalman observation model is $h(p, \nu) = h(p) + \nu$.

The observation models $h^{\text{ES}}$ and $h^{\text{ID}}$ are derived using the projection model described in section 2.3.1 and 2.3.2. ES represents the 3-space point to be projected explicitely, such that the standard projection model can be applied directly. In contrast to this, the landmarks 3-space position has to be determined for ID to allow for applying the projection. The model $h^{\text{PD}}$ is derived using the TFT (see section 2.3.4 and A.1). Thus, no explicite computation of a point in 3-space has to be carried out. The proposed models assume no translation and rotation between the IMU frame $i$ used for prediction and the camera frame $c$ (i. e., $\Gamma_i^n = \Gamma_c^n$). The model including the translation and rotation between camera and IMU is given in appendix A.2.2 and its derivatives in A.3.2.

To ease the readability, some mapping parameters are omitted in the following. For the landmark observation model $h(\pi, \mathcal{X})$ the pose $\pi$ is omitted, since it is the same for all equations. The projection matrices $\mathbf{P}_c$ and $\mathbf{P}_{c'}$ for the master and slave camera respectively are given by the actual navigation prediction ($\Gamma_i^e$ for ES/ID, $\Gamma_i^n$ for PD). For the projection by $\mathbf{P}_{c'}$ the stereo constraints (i. e., $\Gamma_c^{c'}$), discussed in section 2.3.3, are used. The homogeneous point $\mathbf{X}$ represents the landmark $\mathcal{X}$ in $\mathbb{P}^3$ and $\mathbf{x}, \mathbf{y} \in \mathbb{P}^2$ its

projection to the master and slave camera respectively.

**ES Observation** As stated above, the ES observation model results directly from the projective geometry. Thus the homogenous images $\mathbf{x}^{\text{ES}}$ and $\mathbf{y}^{\text{ES}}$ of $\mathcal{X}^{\text{ES}}$ are:

$$\begin{aligned}\mathbf{x}^{\text{ES}} &= \mathbf{P}_c\,\mathbf{X}^{\text{ES}} = \left[\left(\begin{array}{c|c} R_e^c & -R_e^c \cdot t_{ec}^e \end{array}\right)\right] \cdot \left[\begin{pmatrix} \mathcal{X}^{\text{ES}} \\ 1 \end{pmatrix}\right] \\ &= \left[\Phi_c^{eT} \cdot (\mathcal{X}^{\text{ES}} - t_{ec}^e)\right] \end{aligned} \tag{4.2.24}$$

$$\begin{aligned}\mathbf{y}^{\text{ES}} &= \mathbf{P}_{c'}\,\mathbf{X}^{\text{ES}} = \left[\left(\begin{array}{c|c} R_e^{c'} & -R_e^{c'} t_{ec'}^e \end{array}\right)\right] \cdot \left[\begin{pmatrix} \mathcal{X}^{\text{ES}} \\ 1 \end{pmatrix}\right] \\ &= \left[\left(\begin{array}{c|c} R_e^c & -R_e^c t_{ec}^e - t_{cc'}^c \end{array}\right)\right] \cdot \left[\begin{pmatrix} \mathcal{X}^{\text{ES}} \\ 1 \end{pmatrix}\right] \\ &= \left[\Phi_c^{eT} \cdot (\mathcal{X}^{\text{ES}} - t_{ec}^e) - t_{cc'}^c\right] \end{aligned} \tag{4.2.25}$$

Using these homogeneous projections the ES observation model ends up with:

$$h^{\text{ES}}\left(\mathcal{X}^{\text{ES}}\right) = \begin{pmatrix} \mathbf{H}\left(\mathbf{x}^{\text{ES}}\right) \\ \mathbf{H}\left(\mathbf{y}^{\text{ES}}\right) \end{pmatrix} = \begin{pmatrix} \mathbf{H}\left(\left[\Phi_c^{eT} \cdot (\mathcal{X}^{\text{ES}} - t_{ec}^e)\right]\right) \\ \mathbf{H}\left(\left[\Phi_c^{eT} \cdot (\mathcal{X}^{\text{ES}} - t_{ec}^e) - t_{cc'}^c\right]\right) \end{pmatrix} \tag{4.2.26}$$

**ID Observation** The observation model for ID landmarks, proposed by Civera, Montiel and Davison (see [MCD06],[CDM08]), is to be adopted to the notation used in this thesis. Except for recomputation of the landmark's position in 3-space the ID measurement function is deduced similar to ES. The 3-space point $X$ has to be computed using the 6 DOF landmark $\mathcal{X}^{ID}$. Using the notation from (4.2.2) let

$$ray\left(\phi, \psi\right) = \begin{pmatrix} \sin\left(\phi\right) \cdot \cos\left(\psi\right) \\ -\sin\left(\psi\right) \\ \cos\left(\phi\right) \cdot \cos\left(\psi\right) \end{pmatrix} \tag{4.2.27}$$

the unit vector pointing from $\mathcal{C}$ to the position of the 3-space landmark. This way $X$ can be computed by:

$$X \quad = \quad \frac{1}{\rho} \cdot ray\left(\phi, \psi\right) + \mathcal{C} \tag{4.2.28}$$

Finally, the homogenous projections $\mathbf{x}^{\text{ID}}$ and $\mathbf{y}^{\text{ID}}$ of $\mathcal{X}^{ID}$ are:

$$\begin{aligned} \mathbf{x}^{\text{ID}} &= \mathbf{P}_c \left[ \begin{pmatrix} X \\ 1 \end{pmatrix} \right] = \left[ R_e^c \cdot (X - t_{ec}^e) \right] \\ &= \left[ R_e^c \cdot \left( \rho^{-1} \cdot ray\left(\phi, \psi\right) + \mathcal{C} - t_{ec}^e \right) \right] \\ &\overset{(*)}{=} \left[ \Phi_c^{eT} \cdot (ray\left(\phi, \psi\right) + \rho \cdot (\mathcal{C} - t_{ec}^e)) \right] \end{aligned} \tag{4.2.29}$$

$$\begin{aligned} \mathbf{y}^{\text{ID}} &= \mathbf{P}_{c'} \left[ \begin{pmatrix} X \\ 1 \end{pmatrix} \right] = \left[ R_e^{c'} \cdot (X - t_{ec'}^e) \right] \\ &= \left[ R_e^c \cdot (X - t_{ec}^e) - t_{cc'}^c \right] \\ &\overset{(*)}{=} \left[ \Phi_c^{eT} \cdot (ray\left(\phi, \psi\right) + \rho \cdot (\mathcal{C} - t_{ec}^e)) - \rho \cdot t_{cc'}^c \right] \end{aligned} \tag{4.2.30}$$

The equalities marked with $(*)$ follow from the fact, that the homogeneous points are scale invariant. For the singularity at $\rho = 0$ a limiting process $\lim_{\rho \to 0} \mathbf{x}^{\text{ID}}$ and $\lim_{\rho \to 0} \mathbf{y}^{\text{ID}}$ proofs the projections validity even for points at infinity ($\rho = 0$).

Using this definition the measurement function for master and slave observations respectively results in:

$$\begin{aligned} h^{\text{ID}}\left(\mathcal{X}^{\text{ID}}\right) &= \begin{pmatrix} \mathbf{H}\left(\mathbf{x}^{\text{ID}}\right) \\ \mathbf{H}\left(\mathbf{y}^{\text{ID}}\right) \end{pmatrix} \\ &= \begin{pmatrix} \mathbf{H}\left( \left[ \Phi_c^{eT} \cdot (ray\left(\phi, \psi\right) + \rho \cdot (\mathcal{C} - t_{ec}^e)) \right] \right) \\ \mathbf{H}\left( \left[ \Phi_c^{eT} \cdot (ray\left(\phi, \psi\right) + \rho \cdot (\mathcal{C} - t_{ec}^e)) - \rho \cdot t_{cc'}^c \right] \right) \end{pmatrix} \end{aligned} \tag{4.2.31}$$

**PD Observation** When transferring the PD landmarks proposed in this thesis to the new camera poses, the TFT (see section 2.3.4) is used. That is, in contrast to ES and ID the 3-space position of a landmark is not

needed. Instead the stereo correspondence coded in $\mathcal{X}^{\text{PD}}$ (see equation (4.2.4), page 72) is mapped to the new cameras $\mathbf{P_c}$, $\mathbf{P_{c'}}$, defining the predicted projections $\mathbf{x}^{\text{PD}}$ and $\mathbf{y}^{\text{PD}}$.

The navigation state for local navigation (see 4.2.2 and 4.2.7) includes the incremental pose $\Gamma_i^n$. The IMU-camera calibration is assumed to be fixed. That is: $\Gamma_c^i = \Gamma_{c_o}^n$. In this section it is set to identity (see above) and thus: $\Gamma_c^i = \Gamma_{c_o}^n = \left\langle 1^3, 0_3 \right\rangle$. Moreover, the stereo calibration is constant (see 4.1.1). That is: $\Gamma_{c_o}^{c_o'} = \Gamma_c^{c'} = \left\langle 1^3, t_{cc'}^c \right\rangle$. Thus, the incremental pose from the old $c_o, c_o'$ to new $c, c'$ camera frames is:

$$\begin{aligned}
\Gamma_{c_o}^c &= \Gamma_i^c \circ \Gamma_n^i \circ \Gamma_{c_o}^n = \Gamma_i^c \circ \Gamma_n^i \circ \Gamma_c^i && (4.2.32)\\
&= \Gamma_n^c = \langle R_n^c, t_{nc}^n \rangle = \left\langle \Phi_c^{nT}, t_{nc}^n \right\rangle && (4.2.33)\\
\Gamma_{c_o'}^c &= \Gamma_i^c \circ \Gamma_n^i \circ \Gamma_{c_o}^n \circ \Gamma_{c_o'}^{c_o}\\
&= \Gamma_i^c \circ \Gamma_n^i \circ \Gamma_c^i \circ \Gamma_{c'}^c && (4.2.34)\\
&= \Gamma_n^c \circ \Gamma_{c'}^c\\
&= \langle R_n^c, t_{nc}^n \rangle \circ \left\langle 1^3, t_{c'c}^c \right\rangle \overset{(2.2.12)}{=} \langle R_n^c, t_{nc}^n + R_c^n \cdot t_{c'c}^c \rangle\\
&= \left\langle \Phi_c^{nT}, t_{nc}^n - \Phi_c^n \cdot t_{cc'}^c \right\rangle && (4.2.35)
\end{aligned}$$

In the following, keep in mind that $t_{cc'}^c = \begin{pmatrix} b & 0 & 0 \end{pmatrix}^T$ due to the stereo constraints. As reference frame for $\mathcal{X}^{\text{PD}}$ the previous camera frame $c_o$ is used (see 4.2.1, page 72). Using the above equalities, the projection matrices used for the TFT deduced in A.1 are:

$$\Gamma_{c_o}^{c_o} : \mathbf{P} = \left[ 1^3 \middle| 0_3 \right] \qquad \Gamma_{c_o}^{c_o'} : \mathbf{P}' = \left[ 1^3 \middle| t_{cc'}^c \right] = \left[ 1^3 \middle| \begin{pmatrix} b \\ 0 \\ 0 \end{pmatrix} \right] \quad (4.2.36)$$

$$\begin{aligned}
\Gamma_{c_o}^c : \mathbf{P}'' &= \left[ \Phi_c^{nT} \middle| - \Phi_c^{nT} \cdot t_{nc}^n \right] && (4.2.37)\\
\Gamma_{c_o}^{c'} : \mathbf{P}''' &= [R_n^c \,| -R_n^c \cdot (t_{nc}^n - R_i^n \cdot t_{cc'}^c)]\\
&= \left[ \Phi_c^{nT} \middle| -\Phi_c^{nT} \cdot t_{nc}^n + t_{cc'}^c \right] && (4.2.38)
\end{aligned}$$

Using the PD landmark $\mathcal{X}^{\text{PD}} = (x^{\text{PD}}, d)$ for defining

$$x := \begin{pmatrix} x^{\text{PD}} \\ 1 \end{pmatrix} \quad x' := \begin{pmatrix} x^{\text{PD}}(1) - d \\ x^{\text{PD}}(2) \\ 1 \end{pmatrix} \tag{4.2.39}$$

the equations (A.1.1) and (A.1.2), page 218, give:

$$\begin{aligned}
\mathbf{x}^{\text{PD}} &= \left[-x'(3)\cdot b\cdot R_n^c\cdot x - d\cdot x(3)\cdot x'(3)\cdot(-R_n^c\cdot t_{nc}^n)\right] \\
&= \left[d\cdot x(3)\cdot x'(3)\cdot R_n^c\cdot t_{nc}^n - x'(3)\cdot b\cdot R_n^c\cdot x\right] \\
&= \left[R_n^c\cdot(d\cdot t_{nc}^n - b\cdot x)\right] && (4.2.40)\\
&\overset{(*)}{=} \left[\Phi_c^{nT}\cdot(b\cdot x - d\cdot t_{nc}^n)\right] && (4.2.41)\\
\mathbf{y}^{\text{PD}} &= \left[R_n^c\cdot(d\cdot t_{nc}^n - b\cdot x) + d\cdot\begin{pmatrix} bx(3)x'(3) \\ 0 \\ 0 \end{pmatrix}\right] && (4.2.42)\\
&= \left[R_n^c\cdot(d\cdot t_{nc}^n - b\cdot x) + \begin{pmatrix} d\cdot b \\ 0 \\ 0 \end{pmatrix}\right] && (4.2.43)\\
&\overset{(*)}{=} \left[\Phi_c^{nT}\cdot(b\cdot x - d\cdot t_{nc}^n) - \begin{pmatrix} d\cdot b \\ 0 \\ 0 \end{pmatrix}\right] && (4.2.44)
\end{aligned}$$

The sign transformations marked with $(*)$ have been applied using the scale invariance, to visualize the similarity to ES and ID models. Note that in contrast to ES and ID for PD only $\Gamma_c^n = \Gamma_i^n$ and not $\Gamma_c^e$ is used for observation. By this the navigation frame's pose $\Gamma_n^e$ is not observed directly and has no effect on the local estimation. Finally, the measurement function for PD is given by:

$$\begin{aligned}
h^{\text{PD}}(\mathcal{X}^{\text{PD}}) &= \begin{pmatrix} \mathbf{H}(\mathbf{x}^{\text{PD}}) \\ \mathbf{H}(\mathbf{y}^{\text{PD}}) \end{pmatrix} \\
&= \begin{pmatrix} \mathbf{H}\left(\left[\Phi_c^{nT}\cdot(b\cdot x - d\cdot t_{nc}^n)\right]\right) \\ \mathbf{H}\left(\left[\Phi_c^{nT}\cdot(b\cdot x - d\cdot t_{nc}^n) - \begin{pmatrix} db \\ 0 \\ 0 \end{pmatrix}\right]\right) \end{pmatrix}
\end{aligned} \tag{4.2.45}$$

It is worth noticing, that the view ray parametrization $b\cdot x$ for PD is strictly linear. In contrast, ID uses the non linear transformation $ray(\phi, \psi)$ of

spherical coordinates (compare equation (4.2.31)).

### 4.2.7 Relocalization

As was introduced in 4.2.2 on page 76, the local navigation parametrization uses the intermediate navigation frame $n$. It corresponds the IMU frame $i_o$ of the previous update (see figure 4.6, page 76). The local PD landmarks are given relative to the corresponding camera frame $c_o$, coupled with $i_o$ by the fixed calibration $\Gamma_{c_o}^{i_o} = \Gamma_c^i$. Thus, after performing a Kalman Update the new navigation frame $n'$ has to be set to the actual $i$ frame. Thus, beside the global $\Gamma_e^n$ and local $\Gamma_n^i$ poses the PD landmarks, which are linked to $n$, have to be transformed to the new reference system.

The landmarks are transferred to the actual IMU frame $i$ using the observation models (see section 4.2.6 and appendix A.2.2 for the full model respectively). The old PD landmarks $\left(\begin{smallmatrix} x_o^{\mathrm{PD}} \\ d_o \end{smallmatrix}\right)$ master projection is transferred using the master projection observation model:

$$x^{\mathrm{PD}} = \mathbf{H}\left(\left[\Phi_c^{nT} \cdot \left(b \cdot \begin{pmatrix} x_o^{\mathrm{PD}} \\ 1 \end{pmatrix} - d_o \cdot t_{nc}^n\right)\right]\right) \tag{4.2.46}$$

The new disparity is determined by the first components difference of the master and slaves projection observation model:

$$\begin{aligned} d &= P_1 \cdot \mathbf{H}\left(\mathbf{x}_o^{\mathrm{PD}}\right) - P_1 \cdot \mathbf{H}\left(\mathbf{y}_o^{\mathrm{PD}}\right) \\ &= \frac{P_1 \cdot \left(R_n^c \cdot \left(b \cdot \begin{pmatrix} x_o^{\mathrm{PD}} \\ 1 \end{pmatrix} - d_o \cdot t_{nc}^n\right)\right)}{P_3 \cdot \left(R_n^c \cdot \left(b \cdot \begin{pmatrix} x_o^{\mathrm{PD}} \\ 1 \end{pmatrix} - d_o \cdot t_{nc}^n\right)\right)} \\ &\quad - \frac{P_1 \cdot \left(R_n^c \cdot \left(b \cdot \begin{pmatrix} x_o^{\mathrm{PD}} \\ 1 \end{pmatrix} - d_o \cdot t_{nc}^n\right) - \begin{pmatrix} d_o \cdot b \\ 0 \\ 0 \end{pmatrix}\right)}{P_3 \cdot \left(R_n^c \cdot \left(b \cdot \begin{pmatrix} x_o^{\mathrm{PD}} \\ 1 \end{pmatrix} - d_o \cdot t_{nc}^n\right) - \begin{pmatrix} d_o \cdot b \\ 0 \\ 0 \end{pmatrix}\right)} \\ &\overset{(*)}{=} \frac{P_1 \cdot \begin{pmatrix} d_o \cdot b \\ 0 \\ 0 \end{pmatrix}}{P_3 \cdot R_n^c \cdot \left(b \cdot \begin{pmatrix} x_o^{\mathrm{PD}} \\ 1 \end{pmatrix} - d_o \cdot t_{nc}^n\right)} \\ &= \frac{d_o \cdot b}{P_3 \cdot \Phi_c^{nT} \cdot \left(b \cdot \begin{pmatrix} x_o^{\mathrm{PD}} \\ 1 \end{pmatrix} - d_o \cdot t_{nc}^n\right)} \end{aligned} \tag{4.2.47}$$

Where $(*)$ holds, because both homogeneous projections only differ in $(\,-b\cdot d_o\;0\;0\,)$. Thus, the projections to the third components are equal. The transformation given here, corresponds to the reduced observation model. When the full camera to IMU calibration is applied, equations (A.2.16) and (A.2.17) in appendix A.2.2 have to be used for $\mathbf{x}^{\mathrm{PD}}$ and $\mathbf{y}^{\mathrm{PD}}$ respectively.

Following the landmark reinitialization, the relocalization of the coordinate frames is applied. Therefore, $\Gamma^e_n$ and $\Gamma^n_i$ are concatenated to the new global pose $\Gamma^e_{n'}$:

$$n \longrightarrow i \;: \qquad \Gamma^e_{n'} := \Gamma^e_n \circ \Gamma^n_i \overset{(2.2.12)}{=} \langle \Phi^e_n \cdot \Phi^n_i, t^e_{en} + \Phi^e_n \cdot t^n_{ni} \rangle \tag{4.2.48}$$

### 4.2.8 Adjustment

Adjustment techniques allow for an improvement of the estimated filter states using additional information. Mostly, multiple poses corresponding to points in time where the system observed the same place of the environment are related by the respective landmarks. This especially holds for the detection of re-visited locations, what is called loop closing. Various adjustments techniques have been proposed, to exploit such closed loops. They range from complete bundle adjustments (optimizing all poses, landmarks and observations, see [HZ03], [ESN06], [Sib+09]) to reduced 'sparse' models, allowing for larger trajectories and reduced computational effort (see [LA09], [Sze11]). Global navigation parametrizations, as the ones for ES and ID discussed in this thesis, require "connections" between the poses to be adjusted. I. e., visual features are needed, that are matched in succeeding images (feature bundles). The local PD parametrization proposed in this thesis, allows for a simple local adjustment, by using the estimated pose increments solely, i. e., no visual features are involved.

The adjustment model proposed here, uses a simple constraint by an externally determined pose difference $\Delta\Gamma$. This difference describes the relative pose between the two system poses for the revisited place. Methods for loop close detection have been discussed in literature (e. g., see [Wil+09], [CN07]) and are out of the scope of this thesis. Thus, they won't be discussed further. To allow for an evaluation of the proposed adjustment, the loop close positions used in section 7.2.5 are selected manually.

The adjustment model proposed in this thesis uses a constraint $g$ en-

forcing the concatenation of estimated local transformations, to match the determined pose difference $\Delta\Gamma$. Therefore, the estimated transformations $i_{(\mathtt{i}-1)} \rightarrow i_{(\mathtt{i})} =: \Gamma^i_{n\,(\mathtt{i})}$ for $\mathtt{i} = \mathtt{k}...\mathtt{k}+\mathtt{N}$ are stacked to a parameter vector $l$ and are constrained by $g$. Let $\Gamma^i_{n\,(\mathtt{i})} = \left\langle t^n_{ni\,(\mathtt{i})}, \Phi^i_{n\,(\mathtt{i})} \right\rangle$ the estimated local pose for time step $\mathtt{i}$ and $\Delta\Gamma = \langle \Delta t, \Delta\Phi \rangle$ the determined pose difference (see figure 4.7). Using this, the adjustment model is defined by:

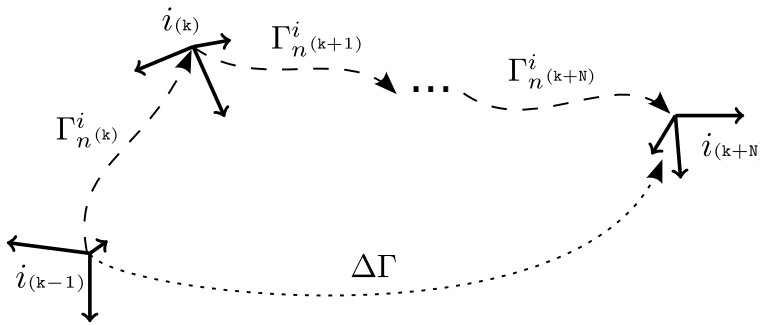


**Figure 4.7.** Local pose transformations and incremental concatenation for final global pose

$$l = \begin{pmatrix} t^n_{ni\,(\mathtt{k})} & \Phi^i_{n\,(\mathtt{k})} & \cdots & t^n_{ni\,(\mathtt{k}+\mathtt{N})} & \Phi^i_{n\,(\mathtt{k}+\mathtt{N})} \end{pmatrix} \tag{4.2.49}$$

$$g_t\,(l) = \Delta t - \sum_{\mathtt{i}=\mathtt{k}}^{\mathtt{k}+\mathtt{N}} \left( \prod_{\mathtt{j}=\mathtt{k}}^{\mathtt{i}-1} \Phi^i_{n\,(\mathtt{j})} \right) \cdot t^n_{ni\,(\mathtt{i})} \tag{4.2.50}$$

$$g_\Phi\,(l) = \begin{pmatrix} 1 - \mathsf{e}_1^{3T} \cdot \left( \prod_{\mathtt{j}=k}^{\mathtt{k}+\mathtt{N}} \Phi^i_{n\,(\mathtt{j})} \right) \cdot \Delta\Phi^T \cdot \mathsf{e}_1^3 \\ 1 - \mathsf{e}_2^{3T} \cdot \left( \prod_{\mathtt{j}=k}^{\mathtt{k}+\mathtt{N}} \Phi^i_{n\,(\mathtt{j})} \right) \cdot \Delta\Phi^T \cdot \mathsf{e}_2^3 \\ 1 - \mathsf{e}_3^{3T} \cdot \left( \prod_{\mathtt{j}=k}^{\mathtt{k}+\mathtt{N}} \Phi^i_{n\,(\mathtt{j})} \right) \cdot \Delta\Phi^T \cdot \mathsf{e}_3^3 \end{pmatrix} \tag{4.2.51}$$

The translation constraint $g_t$ transfers the local position increments $t^n_{ni(\mathtt{i})}$ to the first frame $i_{(\mathtt{k})}$ by:

$$\left(\prod_{\mathtt{j}=\mathtt{k}}^{\mathtt{i}-1} {\Phi^i_n}^T{}_{(\mathtt{j})}\right) \cdot t^n_{ni(\mathtt{i})}$$

Afterwards all translations are concatenated in $i_{(\mathtt{k})}$. The concatenation of all those position increments has to end up in the given translation $\Delta t$. Correspondingly, the orientation constraint $g_\Phi$ concatenates all orientations, what results in $\Delta\Phi$. That is:

$$\prod_{\mathtt{j}=k}^{\mathtt{k}+\mathtt{N}} \Phi^i_{n(\mathtt{j})} \stackrel{!}{=} \Delta\Phi \quad \Longrightarrow \quad \left(\prod_{\mathtt{j}=k}^{\mathtt{k}+\mathtt{N}} \Phi^i_{n(\mathtt{j})}\right) \cdot \Delta\Phi^T \stackrel{!}{=} 1^3$$

Since this is a degenerated matrix constraint (system of 9 equations with rank 3 in total), the constraint is modified. Using the orthonormal basis $\left\{\mathsf{e}^3_1, \mathsf{e}^3_2, \mathsf{e}^3_3\right\}$, the constraint (4.2.51) corresponds to:

$$\begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} - diag\left(\left(\prod_{\mathtt{j}=k}^{\mathtt{k}+\mathtt{N}} \Phi^i_{n(\mathtt{j})}\right) \cdot \Delta\Phi^T\right)$$

The full adjustment constraint $g := \begin{pmatrix} g_t & g_\Phi \end{pmatrix}^T$ is made up of the position constraint $g_t$ as well as the orientation constraint $g_\Phi$. Both can be applied independently. Nevertheless, to exploit the full information on the adjustment parameters, they are used in a single optimization estimation. As can be seen, the position constraint $g_t$ includes the translations $t^n_{ni(\mathtt{j})}$ and the orientations $\Phi^i_{n(\mathtt{j})}$. Thus, an adjustment using this constraint only effects the position and orientation estimation using $\Delta t$ only. In contrast to that, the orientation constraint $g_\Phi$ includes the orientations solely. Using the linearized constraint model given in [MFW04], the error $v$ in $l + v = \bar{l}$, can be estimated (see section 3.2).

## 4.3 Error Modeling

The errors, to be modeled for linear estimation, can be partitioned into two main categories. On the one hand, violations of assumptions on system models induce errors due to linearization or improper parameter constraints. The latter can be caused, by modeling dynamic processes as constant (e. g., constant velocity assumption) or erroneous hardware calibrations. This class of errors is called system errors in the following. Some of these can be addressed, by using augmented system models. Others have to be compensated by noise models. The second class, called sensor errors, is incorporated by sensing noise, that is included in the sensors measurements. This class is addressed by noise models solely.

For the validity of Kalman Filters the tuning of noise parameters is crucial. The covariances, assumed for the prediction and observation process, determine the weights for fusing the state prediction and the gathered observations. This not only effects the estimated state parameters but also the respective covariances. Thus, invalid assumptions on the filters noise model decrease the estimations quality and can cause the filter to diverge. Determining the magnitude of both classes' covariances is subject to filter tuning. Due to the non deterministic character of system errors, their tuning has to be done heuristically. In contrast to this, in most cases sensing noise can be determined by analyzing the sensor output, or are given by hardware specifications.

**PD Locality** It has to be stated, that the local state parametrization of the PD estimator contains an implicit error model. It corresponds to a state augmentation (see 3.4.2, page 58), that eliminates the global poses bias in the update equation. The global state parametrizations ES and ID use the global pose for the filter update models. That is, the errors of all preceeding predictions, i. e., the pose bias, effect the actual estimation. Even though the bias correlations are modeled in the state covariance, small inconsistencies in the prediction have a high impact to these relations. For PD only the local pose is involved in the observation model. By this, only the actual prediction is involved in the actual estimation directly. The global pose is modified indirectly due to the state internal correlations. Due to this decomposition the PD parametrization is more robust to system errors.

### 4.3.1 System Errors

System errors in IMU-aided visual SLAM applications are most often due to simplified movement models, erroneous camera or IMU calibrations and the IMUs hardware limitations. Models to cover these errors are discussed in the following. Because calibrations of internal camera parameters (see section 2.4.1,page 26) can be achieved with high precision, their effects on visual SLAM are marginal. Thus, in especially for perspective cameras with a moderate field of view these are assumed to be negligible.

#### Inertial Prediction

The movement prediction, using inertial models, bears different error sources. On the one hand, integrating in discrete time assumes constant accelerations and turn rates for the respective times lice $\delta\tau$ resulting in integral errors. Thus, linearization errors have to be taken into account when predicting the navigation state. On the other hand, side effects as the earth's rotation and Coriolis force are neglected in the movement models used in this thesis. These errors have to be modeled using the Kalman Filters prediction noise $\omega$ (see section 3.3.1, page 47). In the following, $\omega^\pi$ is used to denote the noisemodeling the pose predictions error.

For the KF the prediction error is taken into account explicitly in the model $f_\mathtt{k}$. Neglecting the sensor noise discussed in section 4.3.2 the integral error model is purely additive, i.e.:

$$f_\mathtt{k}^\pi\left(p, \eta, \omega_\mathtt{k}^\pi\right) = f_\mathtt{k}^\pi\left(\pi, \eta\right) + \omega_\mathtt{k}^\pi \quad \Longrightarrow \quad W_\mathtt{k}^\pi \equiv 1^9 \tag{4.3.1}$$

This additive model is chosen, since the precise relations between the system state and the integral error are either unknown or too complex to model. Thus, this heuristic model is subject to filter tuning (see section 7.2.2, page 172). Nevertheless, assuming the error induced by unmodeled side effects as negligible, an approximation of the integral error can be deduced as described below.

The system errors due to the earth's rotation, Coriolis force and the like can be covered using more advanced inertial movement models (see [Wen07], [TW04]). Since the magnitude of the earth's rotation is known, it can be taken into account. This implies, that the orientation of the earth's

rotation axis relative to the navigation frame is known, or an initial value for estimation is available. Since the proposed SLAM system does not assume such information, these advanced models are not applied here.

**Integral Noise** In the following, $t$, $v$ and $\Phi$ describe the system's position, velocity and orientation respectively for the global and as well as for the local navigation (see section 4.2.5, page 80). Accordingly, $a$ and $w$ give the systems accelerations and turn rates respectively. Since the derivations are very similar for global and local navigation, the frame indices are omitted. Using $t_{\mathtt{k}+1}$ for the position at time $\tau_{\mathtt{k}}$ (for other entities accordingly) and $\delta\tau = \tau_{\mathtt{k}+1} - \tau_{\mathtt{k}}$, the Taylor expansion for the position, velocity and orientation over time gives:

$$\begin{aligned} t_{\mathtt{k}+1} &= \sum_{\mathtt{i}=0}^{\infty} \frac{1}{\mathtt{i}!} \cdot \left.\frac{\partial^{\mathtt{i}} t}{\partial^{\mathtt{i}} \tau}\right|_{\tau_{\mathtt{k}}} \cdot (\tau - \tau_{\mathtt{k}})^{\mathtt{i}} \\ &= t_{\mathtt{k}} + \delta\tau \cdot v_{\mathtt{k}} + \frac{1}{2} \cdot \delta\tau^2 \cdot a_{\mathtt{k}} + \frac{1}{6} \cdot \delta\tau^3 \cdot \left.\frac{\partial a}{\partial \tau}\right|_{\tau_{\mathtt{k}}} + \sum_{\mathtt{i}=4}^{\infty} \frac{\delta\tau^{\mathtt{i}}}{\mathtt{i}!} \left.\frac{\partial^{\mathtt{i}-2} a}{\partial^{\mathtt{i}-2} \tau}\right|_{\tau_{\mathtt{k}}} \end{aligned} \tag{4.3.2}$$

$$v_{\mathtt{k}+1} = v_{\mathtt{k}} + \delta\tau \cdot a_{\mathtt{k}} + \frac{1}{2} \cdot \delta\tau^2 \cdot \left.\frac{\partial a}{\partial \tau}\right|_{\tau_{\mathtt{k}}} + \sum_{\mathtt{i}=3}^{\infty} \frac{1}{\mathtt{i}!} \cdot \left.\frac{\partial^{\mathtt{i}-1} a}{\partial^{\mathtt{i}-1} \tau}\right|_{\tau_{\mathtt{k}}} \cdot \delta\tau^{\mathtt{i}} \tag{4.3.3}$$

$$\Phi_{\mathtt{k}+1} = \Phi_{\mathtt{k}} + \delta\tau \cdot w_{\mathtt{k}} + \frac{1}{2} \cdot \delta\tau^2 \cdot \left.\frac{\partial w}{\partial \tau}\right|_{\tau_{\mathtt{k}}} + \sum_{\mathtt{i}=3}^{\infty} \frac{1}{\mathtt{i}!} \cdot \left.\frac{\partial^{\mathtt{i}-1} w}{\partial^{\mathtt{i}-1} \tau}\right|_{\tau_{\mathtt{k}}} \cdot \delta\tau^{\mathtt{i}} \tag{4.3.4}$$

In the following, the assumptions are made that:

$$\left| \frac{1}{6} \cdot \delta\tau^3 \cdot \left.\frac{\partial a}{\partial \tau}\right|_2 \right| \overset{!}{\gg} \left| \sum_{\mathtt{i}=4}^{\infty} \frac{1}{\mathtt{i}!} \left.\frac{\partial^{\mathtt{i}-2} a}{\partial^{\mathtt{i}-2} \tau}\right| \cdot \delta\tau^{\mathtt{i}} \right|_2 \tag{4.3.5}$$

$$\left| \frac{1}{2} \cdot \delta\tau^2 \cdot \left.\frac{\partial a}{\partial \tau}\right|_2 \right| \overset{!}{\gg} \left| \sum_{\mathtt{i}=3}^{\infty} \frac{1}{\mathtt{i}!} \cdot \left.\frac{\partial^{\mathtt{i}-1} a}{\partial^{\mathtt{i}-1} \tau}\right| \cdot \delta\tau^{\mathtt{i}} \right|_2 \tag{4.3.6}$$

$$\left|\frac{1}{2}\cdot\delta\tau^2\cdot\frac{\partial w}{\partial\tau}\right|_2 \quad \overset{!}{\gg} \quad \left|\sum_{\mathtt{i}=3}^{\infty}\frac{1}{\mathtt{i}!}\cdot\frac{\partial^{\mathtt{i}-1}w}{\partial^{\mathtt{i}-1}\tau}\cdot\delta\tau^{\mathtt{i}}\right|_2 \tag{4.3.7}$$

These assumptions are feasible, since systems used for SLAM undergo a certain inertia. Thus, changes in the accelerations and turn rates derivatives are expected to be moderate. Moreover, the coefficients for the Taylor expansion decrease strongly for the remaining terms. In especially, due to the high sample rate for the IMU $\delta\tau \ll 1$ holds, thus $\delta\tau^{\mathtt{i}}$ decreases rapidly. By this, the error in the prediction models in section 4.2.5, resulting from violation of constant accelerations and turn rates assumptions, can be approximated by

$$\epsilon_p(\tau) \cong \frac{1}{6}\cdot\delta\tau^3\cdot\frac{\partial a}{\partial\tau} \qquad \epsilon_v(\tau) \cong \frac{1}{2}\cdot\delta\tau^2\cdot\frac{\partial a}{\partial\tau} \qquad \epsilon_\Phi(\tau) \cong \frac{1}{2}\cdot\delta\tau^2\cdot\frac{\partial w}{\partial\tau} \tag{4.3.8}$$

for the position, velocity and orientation respectively. Using these approximations, the prediction models process noise due to integration of accelerations and turn rates can be deduced from the physical properties of the hardware system. In the experiments section 7.2.2 (page 182) this way of determining the prediction noise will be analyzed further.

#### Observation Models

The error models for the observations focus on the imaging process and the stereo correspondence to landmark mapping. The former induces deformations and correlations in the used covariance matrices due to pixel normalization. The latter assumes, that the observation model maps the landmarks to the ideal features in the images. That is, the used camera pose $\Gamma_c^e$ has to coincide with the system pose. Since the IMU pose $\Gamma_i^e$ is used in the filter state, the possibly erroneous IMU to camera calibration $\Gamma_c^i$ has to be taken into account.

**Pixel Normalization** Using input data such as feature positions and camera calibrations, pixel normalization (see section 2.4.1) can be applied preceding the initialization and estimation process. Modeling the error of pixel normalization is done, by mapping the input data $y'$ and the according covariance matrix $C_{y'}$ to $y$, $C_y$ in equations (3.3.8) and (3.3.25) respectively

(the time index k is omitted). Given a mapping $g$ this is done by:

$$y = g\left(y'\right) \quad \text{and} \quad C_y = \left.\frac{\partial g}{\partial}\right|_{y'} \cdot C_{y'} \cdot \left.\frac{\partial g}{\partial}\right|_{y'}^{T} \tag{4.3.9}$$

By this, transformations of acquired raw data are used to simplify update models. It is assumed, that the feature $y'$ has been undistorted and rectified using an internal camera calibration, having a negligible error. By this, reversing the camera's $K$-matrix (see section 2.3.2, page 21) is the only deformation on $y'$. For a constant $C_{y'}$ and camera matrix, this results in:

$$y = K^{-1} \cdot y' \quad \text{and} \quad C_y = K^{-1} \cdot C_{y'} \cdot K^{-T} \tag{4.3.10}$$

Thus, the covariance matrix needed for filter update can be determined in advance, once the camera matrix and pixel noise is known.

In [CDM08] Montiel et al. include the image undistortion mapping in their transformation models $g$. This way the deformation of the observations covariance matrix due to image undistortion can be modeled. As will be shown in the experiments section 7.2.4 (page 192), these models have only marginal impact on the estimation for cameras with moderate lens aberration.

Moreover, for high calibration errors the observations stay correlated over time (bias in undistortion). These systematical errors violate the assumption, that observations for distinct updates are stochastically independent. Thus, the systematical error due to erroneous calibrations cannot be modeled this way. Augmenting the system state by the internal camera calibration would take time correlations into account, but the models are highly non linear, and thus their estimation is unlikely to be stable. By this, the assumption of negligible camera calibration errors made above is essential for the proposed systems.

**IMU-Camera Calibration** In contrast to the cameras internal calibration, the calibration of camera and IMU can only be estimated up to a certain quality. On the one hand, this quality is limited by the precision (noise, hardware limits etc.) of the used IMU. Since the influence of those entities cannot be reduced, they have to be taken into account. On the other hand, the

calibration process conditions, as available trajectories and the like (see section 6.9), have a major influence on the calibration's quality.

In contrast to the camera's internal calibration the inclusion of the camera-IMU calibration parameters in the observation models is straight forward (see appendix A.2.2). In addition, an estimated camera-IMU alignment can be used as initialization for the respective parameters. When calibration methods as in section 6.9 are used, the parameters covariance matrix is known. Using this filter augmentation and initialization the correlations in time due to erroneous camera-IMU alignment can be modeled exactly.

#### IMU Models

The bias $b_a$,$b_w$ and scale $s_a$,$s_w$ nuisance parameters (see section 4.2.3) are used, to model correlations in succeeding IMU measurements. The state prediction model used in this thesis assumes these biases and scales to be constant. Even though this is feasible for the scale factors, the biases undergo a certain drift. To compensate for this, this thesis takes the approach, to use a first order random walk error model (similar to [PWA10], [PWA06]). That is, an additive noise $\omega^b$ is used, as was done for the navigation prediction.

The biases' drift is mostly due to environmental influences, such as atmospheric pressure and temperature. Thus, the random walk assumption is imperfect. This leads to the constraint of stable application areas with moderate environmental changes. For typical environments, the biases are in an order of magnitude of $10^{-2}$ in $\left[\frac{m}{s^2}\right]$ for the acceleration and $\left[\frac{rad}{s}\right]$ for the turn rates respectively. For the experiment chapter 7 a drift of $10^{-4}$ $\left[\frac{m}{s^2}\frac{1}{\sqrt{s}}\right]$ for the acceleration and $10^{-5}$ $\left[\frac{rad}{s}\frac{1}{\sqrt{s}}\right]$ for the turn rate biases respectively is assumed. These are approximated empirically and have proven to keep the biases variances from degenerating.

As for cameras, IMUs have the need of internal calibrations (see section 2.4.2, page 29). Beside the modeled IMU nuisance parameters, these are, e. g., the axis misalignments. Due to the implicit compensation by the sensors device drivers, they are considered negligible and are not included in the error models.

Hardware induced errors for IMUs are the hysteresis and the saturation. They arise from applying such sensors under conditions outside of their

hardware specifications, e. g., very low or high accelerations. Since such hardware malfunctions can hardly be detected reliably, they cannot be modeled adequately. Thus, it is assumed in the following, that the sensors hardware limits are held for the proposed SLAM systems.

### 4.3.2 Sensing Errors

The sensing errors in inertial aided SLAM are restricted to the IMU measurements in motion prediction and landmark matching in observation processing by nature. The former are incorporated in the prediction models $f$ (see section 4.2.5), the latter in the observation model $h$ (see section 4.2.6).

#### IMU Prediction

When using sensor data delivered by the IMU, additional noise for the control input has to be used. This direct measurement noise $\omega^a$ and $\omega^w$ can be modeled by zero mean Gaussian noise (see [Wen07]). Knowing the covariances for these, $\omega^a$ and $\omega^w$ can be modeled as additives to the control inputs $a$ and $w$.

That is, in equations (4.2.20)/(4.2.21) on page 80 and equation (4.3.1) noise $\omega^a$ and $\omega^w$ is used in addition to the navigation noise $\omega^\pi$ and the bias drift $\omega^b$. By this, the navigation prediction model including IMU sensing noise is:

$$
\begin{aligned}
f^\pi(\pi, \eta, \omega) \quad = \quad & \begin{pmatrix} t_{ni}^n \\ 0_3 \\ \Phi_i^n \end{pmatrix} + \begin{pmatrix} \delta\tau \cdot v_{ni}^n \\ v_{ni}^n \\ \delta\tau \cdot (s_w \cdot (w + \omega^w) + b_w) \end{pmatrix} \\
& + \begin{pmatrix} \frac{1}{2}\delta\tau^2 \left( s_a \cdot \Phi_i^n (a + \omega^a + b_a) + {\Phi_n^e}^T \cdot \gamma^e \right) \\ \delta\tau \left( \Phi_i^n (s_a \cdot (a + \omega^a) + b_a) + {\Phi_n^e}^T \cdot \gamma^e \right) \\ 0_3 \end{pmatrix} + \omega^\pi
\end{aligned}
\tag{4.3.11}
$$

The respective process noise Jacobian $W^{a,w}$ results in:

$$W^{a,w} = \begin{pmatrix} \frac{1}{2}\delta\tau^2 \cdot s_a \cdot \Phi_i^n & 0^3 \\ \delta\tau \cdot s_a \cdot \Phi_i^n & 0^3 \\ 0^3 & \delta\tau \cdot s_w \cdot 1^3 \end{pmatrix} \tag{4.3.12}$$

The measurement covariances, used for the IMU, are determined by analyzing the sensor output. Therefore, a time interval is chosen, where neither accelerations nor rotations occur. This allows for a computation of the empirical covariances of the respective components. Thus, no heuristic tuning is needed.

#### Observation

As for the integral noise, the observation error is assumed to be additive noise after application of the observation model $h$. Since the observation covariance has been adopted to the pixel normalization deformation (see section 4.3.1), no additional error modeling is needed.

The observation covariance $C_{y'}$ prior to normalization is determined by analyzing the quality of the used SIFT features matcher (see section 2.5, page 34). For a resolution of $640 \times 480$ average values of up to 3 pixels and 2 pixels for $320 \times 240$ were found to be adequate settings. For the reduced resolution down sampling of full resolution images was applied, resulting in an improved signal to noise ratio relative to the image resolution. By this, the relative uncertainty of feature matching is reduced compared to the full resolution images.

During this analysis and later experiments it was observed, that the feature descriptors allow for an appraisement of the features uncertainty. In especially the size (SIFT radius) of the described image patch influences it's matching precision. Solving the task, of how to incorporate this knowledge in the process of determining the observations covariance, could deliver a powerful tool for filter tuning. By this, it is considered as future work.

Chapter 5

# Model Evaluation

To validate the probabilistic properties of the proposed models, the assumed Gaußian distributions $\varphi$ for the modeled entities have to be compared with their true probability distributions $\bar{\varphi}$. Since the analytic distributions can hardly be determined, this is accomplished by using empirical distributions computed by numerical integration in equation (3.1.24), page 41. Since metrics have to be related to the stereo's baseline, they are given in multiples of baselines $[b]$.

As basis the assumption is made, that the used poses $\Gamma$ and the observations of stereo correspondences $x \leftrightarrow y$ undergo a given Gaußian noise. That is:

$$\left( \Gamma \right) \sim \mathcal{N}\left(\bar{\Gamma}, C_\Gamma\right) \tag{5.0.1}$$

$$\begin{pmatrix} x \\ y \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} \bar{x} \\ \bar{y} \end{pmatrix}, C_{2D}\right) \tag{5.0.2}$$

In the following a set $M = \{p_i := (x^i, y^i, \Gamma^i) \,|\, i \in 1...\mathtt{N}\}$ of particles is generated, using the block diagonal matrix

$$C_M = \begin{pmatrix} C_\Gamma & 0 \\ 0 & C_{2D} \end{pmatrix} \tag{5.0.3}$$

as covariance for a Gaußian probability distribution. This set of $\mathtt{N}$ particles $p_i$ (poses $\Gamma^i = \langle \Phi^i, t^i \rangle$ and correspondences $x^i$, $y^i$) is utilized to create ground truth probability distributions $\hat{\varphi}$ for the entities used in the system models, i. e., landmarks, their reprojections and residuals. The number $\mathtt{N}$ of used samples is adopted to the complexity of resulting empirical distributions and used algorithms. To determine a reliable empirical distribution for a

transformation $f$, a **K**ernel **D**ensity **E**stimation (KDE, see [Par62], [DH03]) is used. It is applied to the set $f_{(M)}$ of mapped particles. By this, the sampling noise for the target distribution can be reduced significantly compared to simple histogram computation.

The consistency of the generated entities with their propagated covariances is validated, by determining the difference to the true error distribution and the informativity measure of the particle clouds. This is done using the methods described in section 3.4.2. The empirical covariances $\hat{C}$ for $\hat{\varphi}$ are computed from $f_{(M)}$, for determining the Gaußian distribution closest to the true PDF $\bar{\varphi}$. If not stated explicitly, $C_{2D}$ for the observation correspondences $x$ and $y$ is assumed to be uncorrelated with a standard deviation of 0.005 normalized pixels (typically 3 pixels in $640 \times 480$ images). That is, $C_{2D} = 0.005^2 \cdot 1^4$. For the usage of informativity measures it's values are rounded to integral percentages, since it's numerical precision was found to be $\leqslant 0.1\%$. Moreover, a deviation of $< 0.5\%$ for inliers does not induce a significant reduction of informativity.

## 5.1 Landmark Initialization

The initialization of landmarks is crucial for the estimation process, since it influences all subsequent estimations. As discussed in section 4.2.4 the initialization of PD landmarks is linear. As will be shown, this overcomes inconsistencies introduced by the non linear initialization for ES and ID landmarks. The ES landmarks suffer from high inconsistencies for large point-camera distances and highly uncertain poses. The ID landmarks cover the point-camera distance correctly, but they become inconsistent for highly uncertain poses, i. e., orientations. Since PD landmarks are represented locally, correlations with the system pose are avoided during initialization. By this, the landmark initialization is independent of errors and uncertainties in the used poses.

### 5.1.1 Covariance Propagation and Particle Clouds

The validation of landmark initialization is done, by analyzing the consistency of the covariances, determined as described in section 6.7.2, pages 142ff.

Therefore, the covariances $C_{ES}$, $C_{ID}$ and $C_{PD}$ respectively are extracted from the according propagated state covariances:

$$C_{\mathcal{L}} \;=\; \begin{pmatrix} C_{\Gamma} & C_{\Gamma,\mathcal{L}} \\ C_{\mathcal{L},\Gamma} & C_{\mathcal{L}} \end{pmatrix} \;=\; J^{\mathcal{L}} \cdot C_M \cdot J^{\mathcal{L}^T} \quad \text{for} \quad \mathcal{L} \in \{ES, ID, PD\} \tag{5.1.1}$$

The particle cloud $M$ of initialization states is created as described on page 99. Additionally sample clouds $M^{\mathcal{L}}$ are generated by using the initialization models $f^{\mathcal{L}}_{\text{init}}$ for ES, ID and PD respectively. That is:

$$M^{\mathcal{L}} \;:=\; f^{\mathcal{L}}_{\text{init}}(M) \;=\; \left\{ f^{\mathcal{L}}_{\text{init}}(p_{\mathtt{i}}) \mid p_{\mathtt{i}} \in M \right\} \tag{5.1.2}$$

These sets are used, to determine the true (empirical) probability density functions $\hat{\varphi}_{\mathcal{L}}$ and their covariance $\hat{C}_{\mathcal{L}}$ for the initialization processes.

### 5.1.2 Depth

In the following, the quality of covariance propagation for the landmark to camera distance ($\mathcal{X}^{\text{ES}}(3)$, $\rho$, $d$) is evaluated. In figure 5.1 the empirical probability distributions $\hat{\varphi}$ and their covariances $\hat{C}$ for the point to image plane distances in ES, ID and PD parametrizations are visualized. In addition to that, table 5.1 depicts the informativity (see section 3.4.2, page 55) for the initialized states and the propagated covariances. For checking $C_{\mathcal{L}} \geqslant \hat{C}_{\mathcal{L}}$ for the respective parametrizations, the eigenvalues of $C_{\mathcal{L}} - \hat{C}_{\mathcal{L}}$ are given in table 5.2.

As already discussed in literature (e. g., see [CDM08]), ID covers the true PDF for close and far points, as can be seen in figure 5.1. The transformation of the Gaußian distribution $C_M$ for the pose and the stereo correspondences to the inverse depth $\rho$ is modeled correctly by linear error propagation. This can be concluded from the fact, that the true, the empirical and the propagated PDFs are congruent for all configurations. Moreover, all eigenvalues for $C_{\text{ID}} - \hat{C}_{\text{ID}}$ and the informativity measures are negligible.

In contrast to that, the ES parametrization clearly suffers from Gaußian models for non Gaußian distributions $\hat{\varphi}_{\text{ES}}$ (see top right graph in figure 5.1, blue curve) for distant landmarks. Moreover, the empirical covariance $\hat{C}_{\text{ES}}$ (marked red in graph) of the generated point cloud deviates strongly from the propagated covariance $C_{\text{ES}}$. This can also be concluded from the ESs

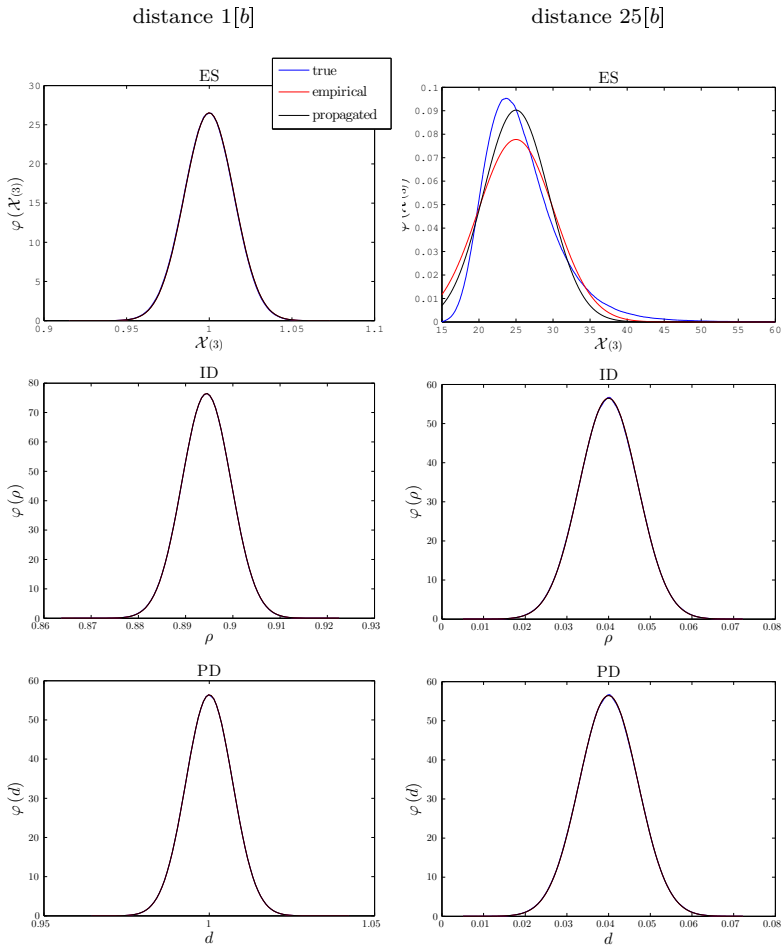


**Figure 5.1.** Probability density functions $\varphi$ of depth components (ES: point to image plane distance $\mathcal{X}^{\mathrm{ES}}{}_{(3)}$; ID: inverse depth $\rho$; PD: disparity $d$). Blue: 'true' PDF $\hat{\varphi}$, generated using sample cloud. Red: 'empirical' Gaußian distribution induced by $\hat{C}$ of sample cloud. Black: distributions for propagated covariances $C_{\mathrm{ES}}$, $C_{\mathrm{ID}}$ and $C_{\mathrm{PD}}$.

**Table 5.1.** Informativity measures $\mathcal{I}\left(M^{\mathcal{L}}, C_{\mathcal{L}}\right)$ for ES, ID and PD initializations for different point to image plane distances. Only the subspaces for $\mathcal{X}^{\text{ES}(3)}$, $\rho$ and $d$ of $M^{\mathcal{L}}$ and $C_{\mathcal{L}}$ were considered.

| | distance 1$[b]$ | | | | distance 25$[b]$ | | | |
|---|---|---|---|---|---|---|---|---|
| | $\mathcal{I}_{0.5}$ | $\mathcal{I}_1$ | $\mathcal{I}_2$ | $\mathcal{I}_3$ | $\mathcal{I}_{0.5}$ | $\mathcal{I}_1$ | $\mathcal{I}_2$ | $\mathcal{I}_3$ |
| ES | 0 | 0 | 0 | 0 | $\mathbf{-4}$ | $\mathbf{-8}$ | $\mathbf{-9}$ | $\mathbf{-5}$ |
| ID | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PD | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| | distance 50$[b]$ | | | |
|---|---|---|---|---|
| | $\mathcal{I}_{0.5}$ | $\mathcal{I}_1$ | $\mathcal{I}_2$ | $\mathcal{I}_3$ |
| ES | $\mathbf{-9}$ | $\mathbf{-19}$ | $\mathbf{-23}$ | $\mathbf{-17}$ |
| ID | 0 | 0 | 0 | 0 |
| PD | 0 | 0 | 0 | 0 |

**Table 5.2.** Eigenvalues of difference of estimated $C$ and empirical $\hat{C}$ covariances $C_{\text{ES}} - \hat{C}_{\text{ES}}$, $C_{\text{ID}} - \hat{C}_{\text{ID}}$ and $C_{\text{PD}} - \hat{C}_{\text{PD}}$.

| | distance 1$[b]$ | | | distance 25$[b]$ | | |
|---|---|---|---|---|---|---|
| | min | max | avg | min | max | avg |
| ES | $-4e^{-7}$ | $1e^{-6}$ | $1e^{-7}$ | $\mathbf{-6.8}$ | $6e^{-6}$ | $\mathbf{-0.8}$ |
| ID | $-5e^{-7}$ | $6e^{-7}$ | $3e^{-8}$ | $-1e^{-5}$ | $1e^{-5}$ | $9e^{-8}$ |
| PD | $-2e^{-7}$ | $3e^{-7}$ | $2e^{-8}$ | $-5e^{-6}$ | $6e^{-6}$ | $8e^{-8}$ |

| | distance 50$[b]$ | | |
|---|---|---|---|
| | min | max | avg |
| ES | $\mathbf{-2e^{6}}$ | $6e^{-6}$ | $\mathbf{-2e^{5}}$ |
| ID | $-1e^{-5}$ | $1e^{-5}$ | $-1e^{-7}$ |
| PD | $-4e^{-7}$ | $6e^{-6}$ | $-1e^{-7}$ |

erroneous inlier ratios (informativity measure $\mathcal{I}$) for distant landmarks in table 5.1. The fact that all components of $\mathcal{I}$ are smaller zero, indicates an inconsistent covariance $C_{\text{ES}}$. This coincides with the appearance of eigenvalues for the covariance difference $C_{\text{ES}} - \hat{C}_{\text{ES}}$ being $\ll 0$, i. e., $C_{\text{ES}} \not\geq \hat{C}_{\text{ES}}$. Thus, ES is not capable to approximate neither the shape of the true PDF, nor it's covariance. This supports the conclusions drawn by Bailey et al. in [Bai+06] (additionally see [CNT04], [Cas+07]).

In contrast to ES, the PD parametrizations covariance covers the distances true probability distribution by the used Gaußian distribution model in disparity. The PDFs non uniform distortion in 3-space due to triangulation is modeled correctly by the disparity representation. As for ID, the PD parametrizations predicted PDF and the empirical PDF in figure 5.1 coincide. Moreover, the eigenvalues for $C_{\text{PD}} - \hat{C}_{\text{PD}}$ and the informativity measures for $C_{\text{PD}}$ with respect to $f_{\text{init}}^{\text{PD}}(M)$ are negligible.

### 5.1.3 View Direction

Beside the landmark distances their view directions are the second important landmark feature. Hence, their initialization's quality is as important for the estimation process as the one for the distances. To the author's knowledge, this issue has not been addressed in literature so far. This is why it is discussed especially exhaustive in the following.

Especially for distant landmarks small errors in the view ray initialization result in a comparably large translational error of the landmark. Thus, wrong error propagation in the rays orientation result in invalid error propagation for the position. Due to this fact the correct propagation of uncertainties and correlations for $\phi$, $\psi$ in ID and $x^{\text{PD}}$ in PD are crucial for consistent landmark estimation. Since ES does not parametrize the view direction itself, the components perpendicular to the depth ($\mathcal{X}^{\text{ES}}(1)$, $\mathcal{X}^{\text{ES}}(2)$ for $\Gamma_e^c = \langle 1^3, 0_3 \rangle$, $x = 0$) are tested.

The validity of $C_{\text{ES}}$, $C_{\text{ID}}$ and $C_{\text{PD}}$, that are generated using initializations with different orientation uncertainties for the pose $\Gamma$, is analyzed in the following. These uncertainties are incorporated to the initialization by setting $C_\Gamma$ accordingly. To be independent of the initialization errors, introduced by the depths uncertainty propagation, only the sub matrices

for the pose and the view directions are considered. That is:

$$C'_{\text{ES}} = \begin{pmatrix} C_\Gamma & C_{\Gamma,\mathcal{X}(1:2)} \\ C_{\mathcal{X}(1:2),\Gamma} & C_{\mathcal{X}(1:2)} \end{pmatrix} \tag{5.1.3}$$

$$C'_{\text{ID}} = \begin{pmatrix} C_\Gamma & C_{\Gamma,\phi} & C_{\Gamma,\psi} \\ C_{\phi,\Gamma} & C_\phi & C_{\phi,\psi} \\ C_{\psi,\Gamma} & C_{\psi,\phi} & C_\psi \end{pmatrix} \tag{5.1.4}$$

$$C'_{\text{PD}} = \begin{pmatrix} C_\Gamma & C_{\Gamma,x^{\text{PD}}} \\ C_{x^{\text{PD}},\Gamma} & C_{x^{\text{PD}}} \end{pmatrix} \tag{5.1.5}$$

For ES, ID and PD respectively. The settings

$$\begin{aligned} p &= \left(0_2, \left(\begin{smallmatrix}1\\0\end{smallmatrix}\right), 1^3, 0_3\right) \\ \rightarrow &\quad x = 0_2\,,\; y = \left(\begin{smallmatrix}-1\\0\end{smallmatrix}\right)\,,\; \Gamma = \left\langle \Phi = 1^3, 0 \right\rangle \end{aligned}$$

having a camera point distance of 1 baseline are used for basic the initialization process. The position is assumed to have standard deviations of 0.05 baselines for each component. After analyzing this special case, the impact of altered initialization setups is discussed.

In table 5.3 the informativity measure is given for standard deviations of 1°, 3°, 5° and 10° for the system's orientation $\Phi$. Accordingly, the differences between the true and the propagated PDFs for the view directions first component are visualized in figure 5.2. As can be seen, the informativity for all parametrizations is valid when using a standard deviation of 1° for the orientation. Moreover, $C_{\text{PD}}$ stays consistent and informative for all initialization scenarios. This is due to the linearity of the PD landmark initialization as introduced in equation (4.2.19), page 79.

For the ES parametrization the propagated covariance $C_{\text{ES}}$ matches the empirically determined $\hat{C}_{\text{ES}}$ for low and medium uncertainties. For standard deviations $\gg 5°$ it tends to be inconsistent ($\mathcal{I} < 0$), even though the informativity is only slightly reduced. Nevertheless, the importance of landmark initializations for the estimation process can cause inconsistent pose estimations. In contrast to ES and PD the ID landmarks become inconsistent even for medium orientation uncertainties. For a standard deviation of 3° measurable inconsistencies between $C_{\text{ID}}$ and $\hat{C}_{\text{ID}}$ (and thus

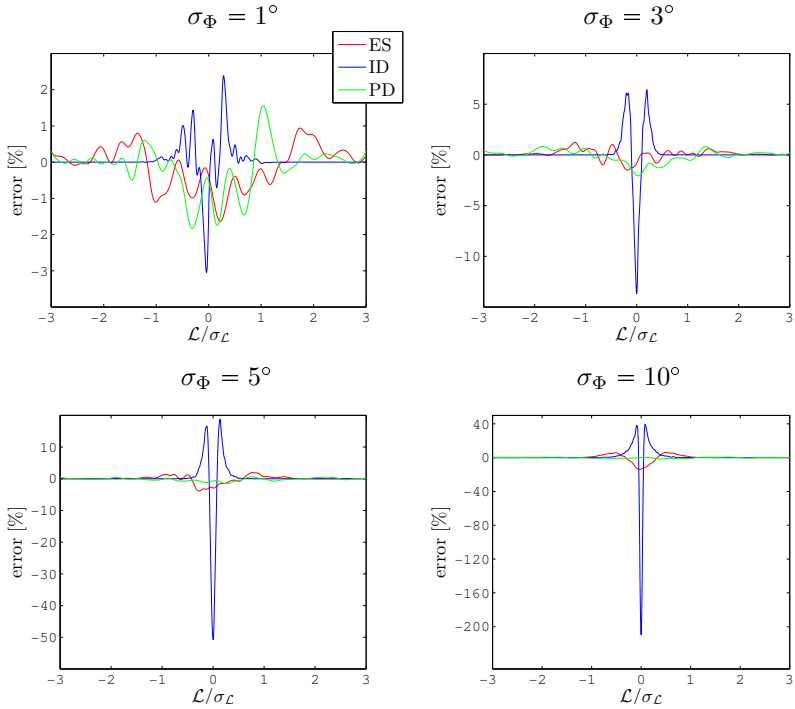


**Figure 5.2.** Error of PDFs in percent of maximum density ($max\,(\hat{\varphi}_\mathcal{L})$). Horizontal axis represents the first component of landmarks view direction ($\mathcal{L} = \left\{ {\mathcal{X}^{\mathrm{ES}}}_{(1)}, \phi, {x^{\mathrm{PD}}}_{(1)} \right\}$) normalized by it's standard deviation. PDFs are cross sections for $\Phi_{(2)} = 0$ of distributions visualized in figure B.1.

**Table 5.3.** Informativity $\mathcal{I}\left(M^{\mathcal{L}}, C'_{\mathcal{L}}\right)$ for sub matrices $C'_{\text{ES}}$, $C'_{\text{ID}}$ and $C'_{\text{PD}}$ with respect to generated point cloud (initialized landmarks). Used initialization setup $\Phi = 1^3$, $x = 0_2$, $y = (-1\ 0)$ (i. e., camera landmark distance 1 baseline). Positions standard deviation 0.05 baselines.

| | $\mathcal{I}_{0.5}$ | $\mathcal{I}_1$ | $\mathcal{I}_2$ | $\mathcal{I}_3$ | $\mathcal{I}_{0.5}$ | $\mathcal{I}_1$ | $\mathcal{I}_2$ | $\mathcal{I}_3$ |
|---|---|---|---|---|---|---|---|---|
| | $\sigma_\Phi = 1°$ | | | | $\sigma_\Phi = 3°$ | | | |
| ES | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| ID | 0 | 0 | 0 | 0 | **−3** | **−4** | −2 | −1 |
| PD | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | $\sigma_\Phi = 5°$ | | | | $\sigma_\Phi = 10°$ | | | |
| ES | 0 | 0 | 0 | 0 | **−4** | **−5** | **−3** | −1 |
| ID | **−12** | **−18** | **−16** | **−9** | **−24** | **−41** | **−48** | **−40** |
| PD | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

$\hat{\varphi}_{\text{ID}}$) occur. As for ES with 10° uncertainty, IDs inconsistency for $\sigma_\Phi = 3°$ is rather small, but can cause an inconsistent estimation due to the repetitive initializations of new landmarks. When increasing the orientations standard deviation to $> 3°$, the inconsistencies increase substantially. Thus, the estimation process for ID parametrizations is likely to become inconsistent, in especially when it is applied to longterm applications. This effect is induced by the violation of the linearization assumptions made for covariance propagation. These non linearities are analyzed on page 108.

Further tests were performed to evaluate the influence of altered point to camera distance and position uncertainty. For better readability the respective graphs and tables have been moved to the appendix.
It was found, that the standard deviation of the position has no impact to the resulting error of propagated covariances for ID and PD parametrizations. This rests on the independence of the view ray initialization to the global position (see equations (4.2.3) and (4.2.4), page 70 respectively). For an experimental validation see table C.1, page 245, in comparison to table 5.3. As can be seen, the informativity measure does not change significantly and can be assumed to be invariant to the positions standard deviation. For validation see also the according figure B.2 on page 235 in comparison to figure 5.2.

In contrast to ID and PD the ESs error in covariance propagation decreases significantly. This is due to the fact, that in ES the view direction and the landmark position are not separated for highly uncertain poses (see above). When the positions uncertainty grows with respect to the orientations uncertainty, the latter becomes negligible. Thus, the results from table 5.3 for low and medium orientation standard deviations can be applied to this experimental setup even for high orientation uncertainties.

Increasing the camera to point distance, strongly effects the consistency of the ES parametrization, as can be seen in table C.2 (page 246) and figure B.3 (page 236). Because in ES the uncertainty of the point to camera distance is highly inconsistent for distant landmarks, the inner correlations of $\hat{\varphi}_{\mathrm{ES}}$ also disturb the view direction. Thus, the consistency of the view ray components cannot be kept. In contrast to this, the ID and PD parametrizations are not effected. On the one hand, the ID landmarks stay inconsistent for high orientation uncertainties with the same amount of non informativity (compare table 5.3). On the other hand, PD landmarks are still consistent and informative. The invariance to landmark distances is founded in the fact, that their initialization is independent of the length of the view ray. For PD this is obvious from equation (4.2.4), page 72. Since the view ray in ID is represented only by the angles of it's spherical coordinates, $sphere\left(|t^e_{cX}|_2^{-1} \cdot t^e_{cX}\right)$ in equation (4.2.3) is independent of the landmark distance.

#### ID Linearization

In the following, the quality of linearization for IDs view ray parametrization $\phi$, $\psi$ is analyzed. Therefore, the linearization error measure $\mathcal{LE}$, defined in section 3.4.3 on page 59, is used.

Using $p = \left(x = 0_2, y = \left(\begin{smallmatrix}-1\\0\end{smallmatrix}\right), \Phi^e_c = 1^3, t^e_{ce} = 0_3\right)$ (as for table 5.3) the Jacobian for $\phi$ is:

$$\left.\frac{\partial \phi}{\partial}\right|_p = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \tag{5.1.6}$$

I. e., 1 for $\partial\Phi^e_{c(2)}$, $\partial x_{(1)}$ and 0 else. Thus, the linearization is only dependent on $\Phi^e_{c(2)}$ and $x_{(1)}$, which correspond to a variation in the images width. Thus,

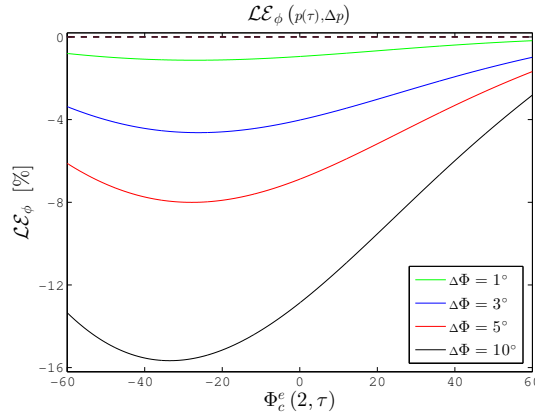


**Figure 5.3.** Linearization error $\mathcal{LE}_\phi$ for initialization of view ray $\phi$ using parameter trajectory $p_{(\tau)}$; Error to propagate : for solid lines $\Delta p = (\Delta x, \Delta y = (0.005, 0.005)\ ,\ (\Delta\Phi, \Delta\Phi, \Delta\Phi)\ ,\ \Delta t = 0_3)$ and $\Delta p = (\Delta x, \Delta y = (0.005, 0.005)\ ,\ (0, \Delta\Phi, 0)\ ,\ \Delta t = 0_3)$ dashed (all $\equiv 0$). First axis represents $\Phi^e_{c(2,\tau)}$ from $p_{(\tau)}$ , $\tau \in [-1..1]$, second axis linearization error.

linear error propagation will introduce only correlations for $\Phi^e_{c(2)}$ and $\phi$ in $C'_{\text{ID}}$ (see equations (5.1.4), page 105 and (5.1.1), page 101). Determining the Hessian for $\phi$ (see equation (A.3.26), page 227) results in:

$$\left.\frac{\partial^2\phi}{\partial\Phi^e_{c(1)}\partial\Phi^e_{c(3)}}\right|_p = 1 \tag{5.1.7}$$

This implies, that in contrast to the second order covariance propagation (e. g., see second order Kalman Filter in [Wen07]) a first order propagation neglects correlations between $\phi$ and $\Phi^e_{c(1)}$ and $\Phi^e_{c(3)}$ respectively. This effects the standard deviation of $\phi$ and the correlations between $\Phi^e_{c(2)}$ and $\phi$. To evaluate the amount of non linearity, the linearization error $\mathcal{LE}_\phi$, as defined in 3.4.3 on page 60, is visualized in figure 5.3. For that, a parameter

trajectory $p_{(\tau)}$ is defined as:

$$p_{(\tau)} \; : \; x \equiv \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \, y \equiv \begin{pmatrix} -1 \\ 0 \end{pmatrix}, \, \Phi^e_{c(\tau)} = \begin{pmatrix} 0 \\ \tau \cdot 60^\circ \\ 0 \end{pmatrix}, \, t^e_c \equiv 0_3 \quad (5.1.8)$$

As can be seen in the figure, neglecting the influence of $\Phi^e_{c(1)}$ and $\Phi^e_{c(3)}$, i. e., setting $\Delta\Phi^e_c = \begin{pmatrix} 0 & \Delta\Phi & 0 \end{pmatrix}$, no linearization error is introduced (dotted lines $\equiv$0). In contrast to that, taking the correlations into account ($\Delta\Phi^e_c = \begin{pmatrix} \Delta\Phi & \Delta\Phi & \Delta\Phi \end{pmatrix}$), the linearization error increases strongly in conjunction with the orientations standard deviation. Note the common relations and signs of these errors compared to the respective covariance inconsistencies, discussed in the previous paragraph (see figure 5.2 and table 5.3). From that it can be concluded, that an important factor for the erroneous covariance propagation is due to the orientation linearization error.

To visualize the impact of the discussed non linearities on covariance propagation, figure 5.4 depicts the difference between the true $\hat{C}_{\text{ID}}$ and the predicted covariance $C_{\text{ID}}$. Although an error of $\sim 1\%$ for the covariance for $\Phi^e_{c(2)}$ and $\phi$ is rather small, the impact to the covariances consistency is not negligible. Using equation (5.1.6) for determining the covariance for $\begin{pmatrix} \Phi^e_{c(2)} & \phi \end{pmatrix}$ results in:

$$C_{\Phi^e_c(2),\phi} = \begin{pmatrix} \sigma^2_\Phi & \sigma^2_\Phi \\ \sigma^2_\Phi & \sigma^2_\Phi + \sigma^2_x \end{pmatrix} \qquad \Phi^e_{c(2)} \sim \mathcal{N}\left(0, \sigma^2_\Phi\right), x_{(1)} \sim \mathcal{N}\left(0, \sigma^2_x\right) \quad (5.1.9)$$

By this, the respective correlation coefficient is

$$r_{\Phi(2),\phi} \overset{(3.1.25)}{=} \frac{\sigma^2_\Phi}{\sqrt{\sigma^2_\Phi} \cdot \sqrt{\sigma^2_\Phi + \sigma^2_x}} = \frac{\sigma_\Phi}{\sqrt{\sigma^2_\Phi + \sigma^2_x}} \quad (5.1.10)$$

being close to 1 for $\sigma_\Phi \gg \sigma_x$, see table 5.4. Thus, even a small change in the correlation causes a loss of consistency, particularly for decreased observation uncertainty (higher image resolution, improved feature detection etc.). Similar results apply for $\Phi^e_{c(1)}$ and $\psi$ (see the respective covariance deviations in figure 5.4).

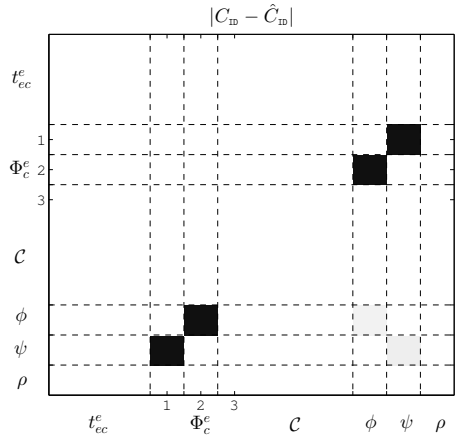


**Figure 5.4.** Absolute difference between $C_{\text{ID}}$ and $\hat{C}_{\text{ID}}$. Note the correlation deviation for $\Phi^e_{c(2)}$, $\phi$ and $\Phi^e_{c(1)}$, $\psi$ respectively. Both are at an order of magnitude of 0.1%, 0.4%, 0.8% and 1.6% for $\sigma_\Phi = 1^\circ$, $3^\circ$, $5^\circ$and $10^\circ$ respectively.

**Table 5.4.** Examples for correlations between orientation and view direction for ID.

| | $\sigma_\Phi$ [rad\|°] | 0.017 \| 1° | 0.052 \| 3° | 0.087 \| 5° | 0.174 \| 10° |
|---|---|---|---|---|---|
| $r_{\Phi(2),\phi}$ | 0.005 | 0.9613 | 0.9955 | 0.9984 | 0.9996 |
| | 0.0025 | 0.9899 | 0.9989 | 0.9996 | 0.9999 |

### 5.1.4 Singularities

Due to the parametrization in spherical coordinates the ID view-ray representation $\phi$, $\psi$ introduces singularities for landmarks above the initialization camera. This is caused by the initialization equation (4.2.17) (page 79). It holds:

$$\begin{aligned} \phi \quad &= \quad \arctan_2\left(\mathtt{x}(1), \mathtt{x}(3)\right) \quad \overset{\mathtt{x}(1)=0}{=} \quad \text{sign}\left(\mathtt{x}(3)\right) \cdot \frac{\pi}{2} \\ &\overset{\mathtt{x}(3)\to 0}{\longrightarrow} \quad \pm 1 \quad \text{contradiction!} \end{aligned} \tag{5.1.11}$$

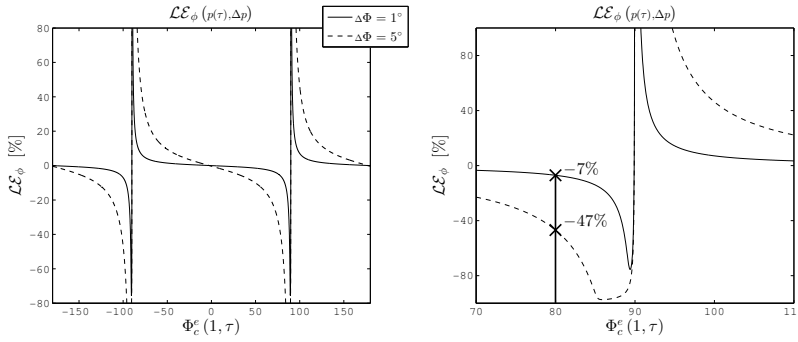


**Figure 5.5.** Linearization error $\mathcal{LE}_\phi$ for initialization of $\phi$ using parameter trajectory $p(\tau)$; First axis represents $\Phi_c^e{}_{(1,\tau)}$ in $p(\tau)$, $\tau \in [-1..1]$, second axis linearization error. Right graph depicts cut-out at the singularity of upward view rays.

Thus, the $\phi$ component of a view-ray perpendicular to the global $\mathtt{e}_2$-axis (up-axis) marks a singularity. This also applies to the Jacobian $J^{ID}$ (equation (A.3.17), page 225):

$$\left.\frac{\partial \arctan_2}{\partial}\right|_{(\mathtt{x}(1),\mathtt{x}(3))} = \left[\left.\frac{\mathtt{x}(3)}{\mathtt{x}(1)^2+\mathtt{x}(3)^2}\right| \frac{-\mathtt{x}(1)}{\mathtt{x}(1)^2+\mathtt{x}(3)^2}\right]$$

$$\overset{\mathtt{x}(1),\mathtt{x}(3)\to 0}{\longrightarrow} \quad [\pm\infty|\pm\infty] \quad \text{contradiction!} \tag{5.1.12}$$

Although a landmark with a view-ray exactly perpendicular to the up-axis is all but impossible, this singularity has major impact on the covariance propagation. For landmarks, initialized with a highly uncertain pose, the covariance cannot be consistent for areas close to the singularity. I. e., for increasing pose uncertainties the inconsistency area grows. Moreover, the landmarks true probability distribution cannot be approximated by a Gaußian distribution.

In figure 5.5 the error of linearization $\mathcal{LE}_\phi\,(p, \Delta p)$ is visualized for the

parameter trajectory.

$$p_{(\tau)} \quad : \quad x \equiv \begin{pmatrix} 0 \\ 0 \end{pmatrix}, y \equiv \begin{pmatrix} -1 \\ 0 \end{pmatrix}, \Phi_c^e{}_{(\tau)} = \begin{pmatrix} \tau \cdot 60^\circ \\ 0 \\ 0 \end{pmatrix}, t_c^e \equiv 0_3$$

$$\Delta p \quad : \quad \Delta x = \Delta y = 0.005\ ,\ \Delta\Phi_c^e = (\Delta\Phi, 0, 0)\ ,\ \Delta t_c^e = 0_3$$

As can be seen, close to the singularity (view-ray parallel to $\mathbf{e}_2$, $\Phi_{c(1)}^e = \pm 90^\circ$) the quality of linearization decreases strongly. Even for $80^\circ$ the error $\mathcal{LE}_\phi$ is at $-7\%$ for $1^\circ$ degree uncertainty and $-47\%$ for $5^\circ$.

## 5.2 Map Representation

To evaluate the consistency of the environment representation, landmark particles are transferred to Euclidian space. Afterwards they are checked for consistency with the true 3-space probability distribution. Therefore, the true pose $\bar{\Gamma}$, stereo correspondence $\bar{x} \leftrightarrow \bar{y}$ and the respective covariance $C_M$ are used to initialize the landmarks $\bar{\mathcal{X}}^{\text{ES}}$, $\bar{\mathcal{X}}^{\text{ID}}$ and $\bar{\mathcal{X}}^{\text{PD}}$ respectively. Accordingly, the covariances $C_{\text{ES}}$, $C_{\text{ID}}$ and $C_{\text{PD}}$ are propagated as in the previous section. Following that, the particle sets $M^{\text{ES}}$, $M^{\text{ID}}$ and $M^{\text{PD}}$ are generated by:

$$M^{\mathcal{L}} := \left\{ \mathcal{L}_{\mathtt{i}} \;\middle|\; \mathcal{L}_{\mathtt{i}} \text{ sample from } \mathcal{N}\left(\bar{\mathcal{L}}, C_{\mathcal{L}}\right) \right\} \qquad \mathcal{L} \in \{\text{ES}, \text{ID}, \text{PD}\} \tag{5.2.1}$$

Note that in contrast to the evaluation of the initialization processes, these clouds are generated in landmark space, but not by an initialization mapping $f_{\text{init}}^{\mathcal{L}}(M)$. Additionally, the set $M$ of poses and stereo correspondences (see page 99) is used to create a 3-space particle cloud $M^{\text{3D}}$ by triangulation, resulting in an empirical probability distribution $\hat{\varphi}^{\text{3D}}$ and the respective covariance $\hat{C}^{\text{3D}}$. Finally, each landmark in the sets $M^{\text{ES}}$, $M^{\text{ID}}$ and $M^{\text{PD}}$ is transferred to 3-space, resulting in $M_{\text{ES}}^{\text{3D}}$, $M_{\text{ID}}^{\text{3D}}$ and $M_{\text{PD}}^{\text{3D}}$. These define the probability distributions $\hat{\varphi}_{\text{ES}}^{\text{3D}}$, $\hat{\varphi}_{\text{ID}}^{\text{3D}}$ and $\hat{\varphi}_{\text{PD}}^{\text{3D}}$ along with their covariances $\hat{C}_{\text{ES}}^{\text{3D}}$, $\hat{C}_{\text{ID}}^{\text{3D}}$ and $\hat{C}_{\text{PD}}^{\text{3D}}$ for ES, ID and PD respectively.

In table 5.5 the eigenvalues for the covariance differences $\hat{C}_{\text{3D}}^{\mathcal{L}} - \hat{C}^{\text{3D}}$ are given, to evaluate consistency, i. e., $\hat{C}_{\text{3D}}^{\mathcal{L}} \geqslant \hat{C}^{\text{3D}}$. For system initialization

the standard setup is used. That is, standard deviation of $x, y$ is 0.005 normalized pixels, for orientation 3° and 0.05 baselines for the position. To maintain comparability for different initialization setups, the given values are in percent of the maximum absolute eigenvalue of $\hat{C}^{\text{3D}}$. As can be seen, all representations deliver a consistent map for camera to point distances of $\leqslant 5$ baselines.

As for the initialization analysis increasing the landmark distance leads to an inconsistent representation in ES parameter space. All eigenvalues of the difference $\hat{C}^{\text{ES}}_{\text{3D}} - \hat{C}^{\text{3D}}$ are negative. Thus, ES is inconsistent in all spatial components and, by this, the worst case $\hat{C}^{\text{ES}}_{\text{3D}} < \hat{C}^{\text{3D}}$ is met. In contrast to that, the ID and PD representations are consistent and informative.

In table 5.6 the same experiment was carried out, with an altered orientation uncertainty of 10°. Going in common with the results discussed in [Bai+06] and [SLP07], the magnitude of the eigenvalues for $\hat{C}^{\text{ES}}_{\text{3D}} - \hat{C}^{\text{3D}}$ imply, that even for short distances ES becomes inconsistent for high orientation uncertainties. Although the average of eigenvalues is $> 0$, one is negative. That is, only parts of the ES parametrization are inconsistent. The remaining are non informative but consistent. Increasing the landmarks distance, the same inconsistency as for 3° standard deviation in the orientation occurs. This is due to the high curvature of the resulting 3-space cloud for close landmarks and high orientation uncertainties.

The clouds curvature results from the fact, that a high variation in orientation causes a point cloud deformation along a circle with the radius of the camera to point distance. The corresponding probability distributions are visualized in figure 5.6. The iso-probability ellipses (confidence intervals) for the $\mathsf{e}_1/\mathsf{e}_3$-subspace of the covariance for ES parameters are marked black. The remaining curved lines depict the iso-probability lines for the true probability density. On the one hand, $\hat{C}_{\text{ES}}$ is able to cover the true distribution in the $\mathsf{e}_1$-plane (parallel to image plane). On the other hand, the distance to the image plane is not modeled properly. The circles front view (along the $\mathsf{e}_3$-axis) corresponds to an ellipse, thus, it's error can be modeled as Gaußian. In contrast to that, the top view (along the $\mathsf{e}_2$-axis) appears similar to a circle segment, which cannot be modeled as Gaußian noise in Euclidian space.

Note that the ID representation models the 3-space distribution correctly, even for high orientation uncertainties. This does not contradict the

**Table 5.5.** Eigenvalues of $\hat{C}^{\mathcal{L}}_{3D} - \hat{C}^{3D}$ in percent of $max\left(|eig(\hat{C}^{3D})|\right)$. Used standard deviations for initialization: position 0.05 $[b]$; orientation 3°; observations 0.005 [npx]

| | 5$[b]$ | | | 10$[b]$ | | | 25$[b]$ | | |
|---|---|---|---|---|---|---|---|---|---|
| | min | max | avg | min | max | avg | min | max | avg |
| ES | −0.4 | 0.1 | −0.2 | **−3.6** | −0.4 | **−1.6** | **−25.9** | −0.7 | **−9.1** |
| ID | −0.1 | 0.1 | 0.0 | −0.1 | 0.0 | 0.0 | −0.3 | 0.0 | −0.1 |
| PD | −0.1 | 0.1 | 0.0 | 0.0 | 0.1 | 0.0 | −0.2 | 0.0 | −0.1 |

**Table 5.6.** Eigenvalues of $\hat{C}^{\mathcal{L}}_{3D} - \hat{C}^{3D}$ in percent of $max\left(|eig(\hat{C}^{3D})|\right)$. Used standard deviations for initialization: position 0.05 $[b]$; orientation 10°; observations 0.005 [npx]

| | 1$[b]$ | | | 10$[b]$ | | | 25$[b]$ | | |
|---|---|---|---|---|---|---|---|---|---|
| | min | max | avg | min | max | avg | min | max | avg |
| ES | **−2.7** | **5.6** | **1.9** | **−2.6** | **4.3** | **1.1** | **−23.3** | **−3.6** | **−11.0** |
| ID | −0.1 | 0.1 | 0.0 | 0.0 | 0.1 | 0.0 | −0.1 | 0.1 | 0.0 |
| PD | −0.1 | 0.0 | 0.0 | −0.1 | 0.0 | 0.0 | −0.2 | 0.1 | −0.1 |

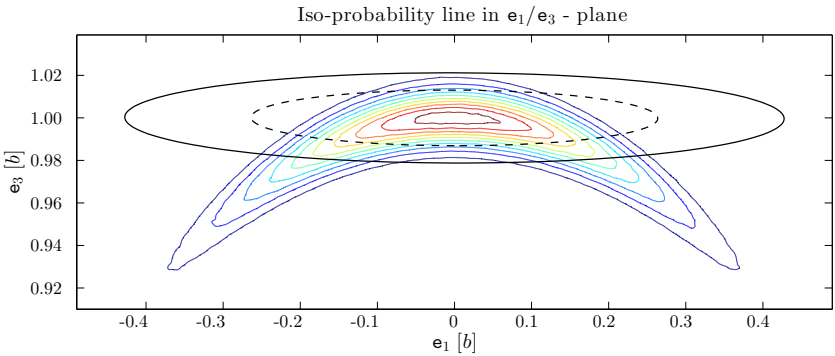


**Figure 5.6.** Distribution of 3D-points in the $\mathbf{e}_1/\mathbf{e}_3$-plane initialized with standard deviation 10° for orientation. Curvatures mark true iso-probability lines, black ellipses confidence intervals for $C_{ES}$ (dotted 68%, solid 95%).

inconsistencies detected for the respective initialization of $\phi$ and $\psi$ analyzed on page 108. Using the true pose $\bar{\Gamma}$ and stereo correspondence $\bar{x} \leftrightarrow \bar{y}$ the initialization is locally valid. Using noisy poses and correspondences far from the true values a valid linearization within the uncertainty area is required but not given. This leads to the conclusion, that only the initialization process in ID is erroneous but not the representation itself.

## 5.3 Landmark Observation

In the following, the validity of error propagation within the observation models $h^{\mathcal{L}}$ for ES, ID and PD respectively (see section 4.2.6, pages 82ff) is analyzed. As in the previous section a ground truth pose $\bar{\Gamma}$ and a stereo correspondence $\bar{x} \leftrightarrow \bar{y}$ are used, to create ground truth landmarks $\bar{\mathcal{X}}^{\text{ES}}$, $\bar{\mathcal{X}}^{\text{ID}}$ and $\bar{\mathcal{X}}^{\text{PD}}$ for the respective parametrizations. Additionally, the pose-landmark covariances $C_{\text{ES}}$, $C_{\text{ID}}$ and $C_{\text{PD}}$ are determined by linear error propagation. These are used to create particle clouds $M^{\text{ES}}$, $M^{\text{ID}}$ and $M^{\text{PD}}$:

$$M^{\mathcal{L}} = \left\{ \begin{pmatrix} \Gamma_{\mathtt{i}} \\ \mathcal{L}_{\mathtt{i}} \end{pmatrix} \middle| \begin{pmatrix} \Gamma_{\mathtt{i}} \\ \mathcal{L}_{\mathtt{i}} \end{pmatrix} \sim \mathcal{N}\left( \begin{pmatrix} \bar{\Gamma} \\ \bar{\mathcal{L}} \end{pmatrix}, C_{\mathcal{L}} \right), \mathtt{i} \in \{1 \dots \mathtt{N}\} \right\} \quad (5.3.1)$$

### Prediction

To simulate pose changes after the landmark initialization, noisy pose increments are added to the particles of all sets. This corresponds to the use of a linear prediction model $f_{\text{pred}}$ with a certain prediction noise $C_\omega$. Finally, the observation models $h^{\mathcal{L}}$ are applied to the predicted particle sets $f_{\text{pred}}(M^{\mathcal{L}})$ resulting in clouds of stereo correspondences:

$$\left\{ \begin{pmatrix} x_{\mathtt{1}}^{\mathcal{L}} \\ y_{\mathtt{1}}^{\mathcal{L}} \end{pmatrix} \dots \begin{pmatrix} x_{\mathtt{N}}^{\mathcal{L}} \\ y_{\mathtt{N}}^{\mathcal{L}} \end{pmatrix} \right\} = M_{\text{2D}}^{\mathcal{L}} := h^{\mathcal{L}}\left( f_{\text{pred}}(M^{\mathcal{L}}) \right) \quad (5.3.2)$$

According to the process of Kalman Filtering (see section 3.3) the covariances $C_{\mathcal{L}}$ are propagated using the Jacobians of $f_{\text{pred}}$ and $h^{\mathcal{L}}$ as well as the prediction noise $C_\omega$. The resulting covariances $C_{\text{2D}}^{\mathcal{L}}$ determine the predicted PDFs $\varphi_{\text{2D}}^{\mathcal{L}}$ of the particles in $M_{\text{2D}}^{\mathcal{L}}$.

### Evaluation

The reference cloud $M_{2D}$ of reprojected stereo correspondences is generated using the set $M$ (noisy poses and correspondences), given on page 99. Each particle $(\Gamma_i, x_i, y_i)$ is triangulated to $(\Gamma_i, X_i)$. After application of the prediction model $f_{pred}$ with the same noisy pose increments as above, the particles $X_i$ are projected to the predicted camera pose $\Gamma'_i$. By this, $M_{2D}$ determines the reprojections empirical PDF $\hat{\varphi}_{2D}$ and it's covariance $\hat{C}_{2D}$.

For the analysis of linearization errors $\mathcal{LE}$ the absolute values $|\mathcal{LE}_i|$ are averaged over all particles $i$. In contrast to the standard linearization error the average is computed for particles being spread in parameter space. Thus, the absolute linearization error is used, to avoid annihilation of opposite signed errors. Let:

$$p_i := f_{pred}\left(\begin{pmatrix} \Gamma_i \\ \mathcal{L}_i \end{pmatrix}\right) \quad \bar{p} := f_{pred}\left(\begin{pmatrix} \bar{\Gamma} \\ \bar{\mathcal{L}} \end{pmatrix}\right) \quad \Delta p_i := p_i - \bar{p} \quad p := \begin{pmatrix} p_1 \\ \vdots \\ p_N \end{pmatrix}$$

corresponding to the error of the mapped particle. Using these definitions the modified linearization error results in:

$$\tilde{\mathcal{LE}}_{h^{\mathcal{L}}}(\bar{p}, p) := \frac{1}{N} \cdot \sum_{i=1}^{N} |\mathcal{LE}_{h^{\mathcal{L}}}(\bar{p}, \Delta p_i)|$$

The informativity measure for $h^{\mathcal{L}}$ is determined by evaluating $\mathcal{I}\left(M_{2D}, C_{2D}^{\mathcal{L}}\right)$. This represents the informativity of the covariance $C_{2D}^{\mathcal{L}}$ for the respective landmarks observation prediction with respect to the true error distribution.

### The Trajectory

The trajectory used for prediction simulation, corresponds to a circle segment centered at the observed landmark. See figure 5.7 for a visualization of the path for a circle segment of 90° ($\frac{1}{4}$-circle). Beside the path the generated 3-space point cloud $(\Gamma_i, X_i)$ is depicted. Using such circle trajectories the impact of changes in the sighting angles after landmark initialization can be simulated. This is especially of interest, since these cause most of the deformation of the predicted PDFs, as can be seen in figure 5.7. To evaluate

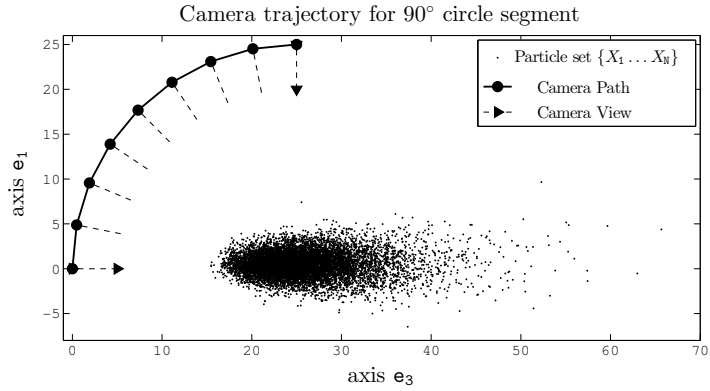


**Figure 5.7.** Camera path in $e_3$-/$e_1$-plane and generated 3-space particle cloud $\{X_1 \cdots X_N\}$ for a distance of 25 baselines and 8 prediction steps. The path starts (LM initialization) at $(0, 0)$ and ends at $(25, 25)$, including 8 pose prediction steps.

**Table 5.7.** Informativity measure $\mathcal{I}$ of observation prediction. Standard deviations: initial position 0.05 $[b]$, orientation 3°; process noise (for each of the 8 prediction steps): position $\sigma_t = 0.01[b]$, orientation $\sigma_\Phi = 0.5°$.

| | distance $1[b]$ | | | | distance $25[b]$ | | | |
|---|---|---|---|---|---|---|---|---|
| | $\mathcal{I}_{0.5}$ | $\mathcal{I}_1$ | $\mathcal{I}_2$ | $\mathcal{I}_3$ | $\mathcal{I}_{0.5}$ | $\mathcal{I}_1$ | $\mathcal{I}_2$ | $\mathcal{I}_3$ |
| ES | 0 | 0 | 0 | 0 | $-3$ | $\mathbf{-13}$ | $\mathbf{-18}$ | $\mathbf{-11}$ |
| ID | 0 | 0 | 0 | 0 | **11** | **6** | $-3$ | $-2$ |
| PD | 0 | 0 | 0 | 0 | 0 | 0 | $-2$ | $-1$ |

the influence of prediction linearization, different numbers of pose increments are used.

### 5.3.1 90° Circle Segment

In table 5.7 the informativity $\mathcal{I}$ for the respective parametrizations is given

**Table 5.8.** Mean of absolute linearization errors $\tilde{\mathcal{LE}}$ for master $x$ and slave $y$ observation predictions. Test setup is the same as in table 5.7.

| | distance 1[$b$] | | | |
|---|---|---|---|---|
| | $\tilde{\mathcal{LE}}_{x(1)}$ | $\tilde{\mathcal{LE}}_{x(2)}$ | $\tilde{\mathcal{LE}}_{y(1)}$ | $\tilde{\mathcal{LE}}_{y(2)}$ |
| ES | 0.1 | 0 | 1.7 | 0 |
| ID | 1.3 | 0 | 2.5 | 0 |
| PD | 0.6 | 0 | 0.1 | 0 |

| | distance 25[$b$] | | | |
|---|---|---|---|---|
| | $\tilde{\mathcal{LE}}_{x(1)}$ | $\tilde{\mathcal{LE}}_{x(2)}$ | $\tilde{\mathcal{LE}}_{y(1)}$ | $\tilde{\mathcal{LE}}_{y(2)}$ |
| ES | 0.4 | 0.4 | 0.4 | 0.4 |
| ID | **21.1** | 0.4 | **21.1** | 0.4 |
| PD | **5.3** | 0 | **5.3** | 0 |

after applying 8 prediction steps (see figure 5.7). As can be seen, all parametrizations are able to cover their true PDF $\hat{\varphi}$ of the reprojected particle cloud $M_{2D}$ for a distance of 1 [$b$] (see figure 5.8). Increasing the distance results in a degraded informativity, i. e., for ES and ID parametrization.

The ES representation ends up with an inconsistently predicted covariance, i. e., $C_{2D}^{ES} < \hat{C}_{2D}$. Thus, the predicted observations are assumed as too certain. For the Kalman Filter this results in an increased weighting of the predicted observations with respect to the measurements. For ID the significant deviations of informativity are all $> 0$. Thus, the informativity is degraded, but no inconsistency is introduced. Nevertheless, the reprojections of landmarks are assumed, to be more uncertain as they actually are. In contrast to ES this causes the observed stereo correspondences to gain a too high weighting during the estimation process. This potentially results in an erroneous estimation. For PD the deviations between the true covariance $\hat{C}_{2D}$ and the predicted $C_{2D}^{PD}$ are negligible. Thus, PD provides a valid observation covariance prediction for the Kalman Filter.

Table 5.8 shows the average linearization error $\tilde{\mathcal{LE}}$ for all components of the reprojection models $h^{\mathcal{L}}$. As for the informativity measure, the linearization errors for all representations are negligible for small camera to landmark distances. This is due to the fact, that for small distances the

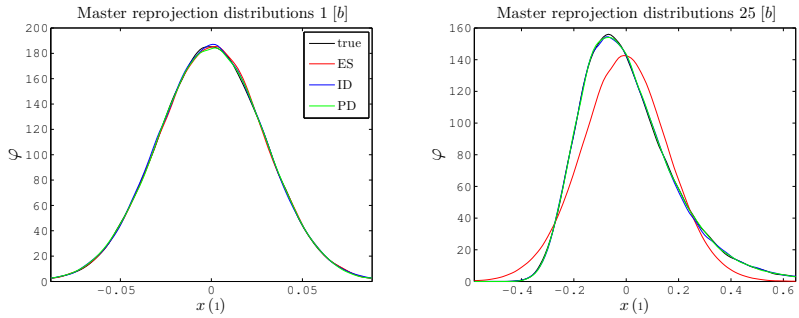


**Figure 5.8.** True $\hat{\varphi}_{2D}$ reprojection probability density function and empirical $\varphi^{\mathcal{L}}_{2D}$ PDFs, for reprojections after $\frac{1}{4}$-circle trajectory. Distributions are cross-sections of the two dimensional distributions for the master projection $x$ at $x_{(2)} \equiv 0$ in the master image. Test setup is the same as in table 5.7.

reprojections distributions are Gaußian (see left graph in figure 5.8). Thus, the PDF-mapping from parameter space to the reprojections is linear.

In contrast to this PD and ID both undergo a high linearization error for a camera to landmark distance of 25 baselines. As can bee seen in the right graph of figure 5.8, the reprojections PDF $\hat{\varphi}_{2D}$ is not Gaußian (note the asymmetry). Since the empirical distributions for ID and PD match $\hat{\varphi}_{2D}$, the linearization error can be considered, to be incorporated by the deformation of the Gaußian landmark noise. This effect is decreased for PD compared to ID due to the repetitive relocalization (see section 4.2.7, page 86) of the filter state. For the 8 prediction steps the relocalization is performed resulting in a piecewise linearization for the intermediate poses. By this, the linearization error for the final observation prediction is reduced.

As can be seen in the right of figure 5.8, even for distant landmarks the ESs PDF transforms to a Gaußian distribution. Thus, the linearization error in the ES representation is negligible. The ellipsoid in 3-space, corresponding to $C_{ES}$, is mapped to an ellipse in the image plane. This is due to the fact, that it is projected along one of it's principal components, after traveling the 90° circle segment. On the one hand, this is a valid mapping for short distances, since the ESs PDF is valid (see previous section) and it's reprojection is Gaußian. On the other hand, high camera landmark distances

**Table 5.9.** Mean of absolute linearization errors $\mathcal{LE}$ for master $x$ and slave $y$ observation predictions. For the test a single prediction step (for 90° circle segment) is used. See table 5.8, for a comparison to 8 prediction steps.

| | distance 1[$b$] | | | |
|---|---|---|---|---|
| | $\mathcal{LE}_{x(1)}$ | $\mathcal{LE}_{x(2)}$ | $\mathcal{LE}_{y(1)}$ | $\mathcal{LE}_{y(2)}$ |
| ES | 0.2 | 0.2 | 0.3 | 0.2 |
| ID | 2.3 | 0 | 1.2 | 0 |
| PD | 0.2 | 0 | 1.2 | 0 |

| | distance 25[$b$] | | | |
|---|---|---|---|---|
| | $\mathcal{LE}_{x(1)}$ | $\mathcal{LE}_{x(2)}$ | $\mathcal{LE}_{y(1)}$ | $\mathcal{LE}_{y(2)}$ |
| ES | 0.4 | 0.1 | 0.4 | 0.1 |
| ID | **27.6** | 0.2 | **27.6** | 0.2 |
| PD | **27.7** | 0 | **27.7** | 0 |

cause an invalid PDF for the ES landmark. Due to this, the reprojected PDF is likewise invalid, even though the linearization error is negligible. Thus, the degraded informativity of the ES landmarks observations are resided in the invalid landmark PDF rather than in the covariance propagation for the observation model $h^{\text{ES}}$.

**Relocalization Impact** As mentioned above, the linearization error for PD landmarks is decreased compared to ID due to the relocalization after each prediction step. Performing only a single prediction step, i. e., no relocalization, on the $\frac{1}{4}$ circle the linearization error of PD and ID are the same, see table 5.9. Comparing tables 5.8 and 5.9 it follows, that the use of a piecewise linearization by relocalization improves the observation prediction for PD. This leads to the conclusion, that adopting the number of relocalizations can help to improve the performance of the PD observation model. Note that the linearization error for ID is also increased slightly. This is due to the fact, that the piecewise linearization of the system poses prediction improves the error propagation especially for the orientation.

**Table 5.10.** Mean of absolute linearization errors $\mathcal{LE}$ for master $x$ and slave $y$ observation predictions as in table 5.8. Traveled trajectory between initialization and projection is a circle segment of 45° (in 4 prediction steps), landmark camera distance is 25 baselines.

| | informativity | | | | linearization error | | | |
|---|---|---|---|---|---|---|---|---|
| | $\mathcal{I}_{0.5}$ | $\mathcal{I}_1$ | $\mathcal{I}_2$ | $\mathcal{I}_3$ | $\mathcal{LE}_{x(1)}$ | $\mathcal{LE}_{x(2)}$ | $\mathcal{LE}_{y(1)}$ | $\mathcal{LE}_{y(2)}$ |
| ES | 1 | −2 | **−4** | −2 | **17.5** | 0 | **17.4** | 0 |
| ID | **4** | **4** | 0 | 0 | 4.2 | 0.3 | 4.4 | 0.3 |
| PD | 0 | 0 | 0 | 0 | 2.8 | 0 | 2.8 | 0 |

### 5.3.2 45° Circle Segment

Shortening the traveled trajectory to a circle segment of 45° inverts the relations in linearization errors. In table 5.10 the respective linearization errors and informativity measures are given in comparison to table 5.7 and 5.8.

For the lower angle between the initialization and observation view the ID and PD representations cover the resulting reprojections PDF much better. In [CDM08], Montiel et al. addressed this phenomenon for the ID parametrization. They defined a linearity measure, similar to $\mathcal{LE}$, to evaluate the relation between the inverse depths uncertainty and quality of landmark reprojection. They found, that a large angle between the reprojections view direction and landmarks initializations ray direction has a major impact on the measurement models linearization quality. That is, a high inverse depth uncertainty and a high angle difference cause significant linearization errors.

Comparing the tests for the relocalization impact (see previous section) and the 45° circle segment, proves a strong connection between the angle difference and the linearization quality. Thus, even though their linearization measure does not take pose uncertainties into account, the results can be assumed to match the observations made here. Moreover, since the geometry of view and landmark direction is similar for ID and PD for the first observation (i. e., no relocalization applied yet), the results can be transferred to the PD representation.

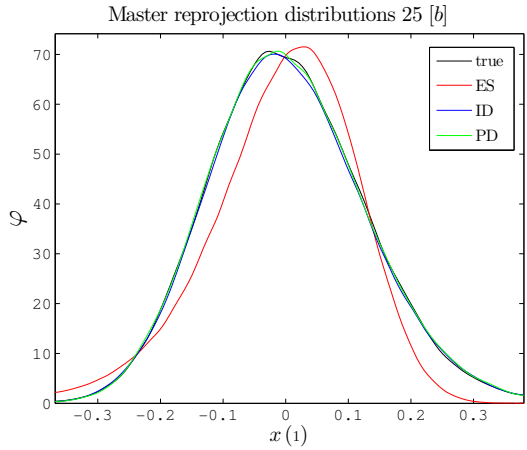


**Figure 5.9.** True $\hat{\varphi}_{2D}$ reprojection probability density function and empirical PDFs $\varphi^{\mathcal{L}}_{2D}$, for reprojections after $\frac{1}{8}$-circle trajectory, divided into 4 prediction steps. Distributions are cross-sections of four dimensional distributions for image row $x_{(2)} \equiv 0$ in the master image.

In contrast to ID and PD, the ES parametrization undergoes a high linearization error. This is due to the fact, that the 3-space ellipsoid, corresponding to $C_{ES}$, is perspectively distorted by the altered view angle. Since the projection is not along one of it's principal components, it's mapping is no ellipse (i. e., not Gaußian). This effect is also visible when comparing the empirical PDFs for $h^{\mathcal{L}}(\mathcal{L}_i)$ to the true PDF for $h(M_{2D})$ (see figure 5.9).

## 5.4 Relocalization

Beside for the influence to observation prediction the PD representations relocalization process (see section 4.2.7, page 86) has to be discussed separately. It is performed periodically, to update the local state representation. The relocalization has to be applied, at least when new landmarks are to be incorporated to the system state. This is due to the fact, that new landmarks

are initialized relative to the pose, they were detected in. It's advantage is, that the observations linearization error is reduced (see previous section). Moreover, it allows for the local pose parametrization, which can be used e. g., for simple adjustment constraints (see section 4.2.8).

A drawback of this necessity of reprojecting the landmarks to the new pose is, that multiple relocalizations without a filter update accumulate the linearization error. Since the amount of future pose increments is not known, determining the best point in time for relocalization is a difficult task. For the filter implementation, described in chapter 6, it is performed for each captured image. Note that this does not exclude images, for which no visual features were detected. This has proven to be convenient, since the cameras frame-rate is sufficient high, to allow for a valid initialization. Moreover, it is ensured, that new landmarks can be incorporated to the filter at time of their occurrence.

A further drawback of relocalization is the possible violation of prerequisites for the trifocal tensor (see section 2.3.4), when landmarks leave the cameras viewport. On the one hand, a prediction of a negative disparity (point behind camera) does not conflict the model assumptions. That is, the landmark represents the same point in 3-space using a negative disparity. It can be kept in state, to allow for a possible re-detection in future images, i. e., the disparity is known to be $\geqslant 0$. On the other hand, singularities occur for landmarks in the camera plane, for which the disparity is not defined.

### 5.4.1 Linearization and Singularities

Landmarks, that are out of the cameras viewport, can have a view ray parallel to the image plane of the new camera, i. e., being located in it's camera plane. A landmark crossing this plane projects with a disparity increasing to infinity in front and decreasing from negative infinity behind the plane (see figure 5.10). This ends up with an undefined disparity within the plane. This is due to the fact, that the stereo correspondences $x \leftrightarrow y$ degenerate to lines. Even though such a view ray is all but impossible, close to parallel rays are instable in linearization.

In figure 5.11 the relocalization's linearization error in the disparity for a landmark close to the camera plane is given. The discussed function $f$ maps the actual filter state $p$ to the disparity of the relocalized landmark.

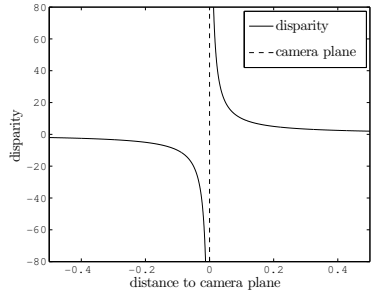


**Figure 5.10.** Disparity for a stereo correspondence in dependence on the minimal distance (i. e., in $\mathbf{e}_3$) to camera plane. For locations within the camera plane, the disparity is undefined.

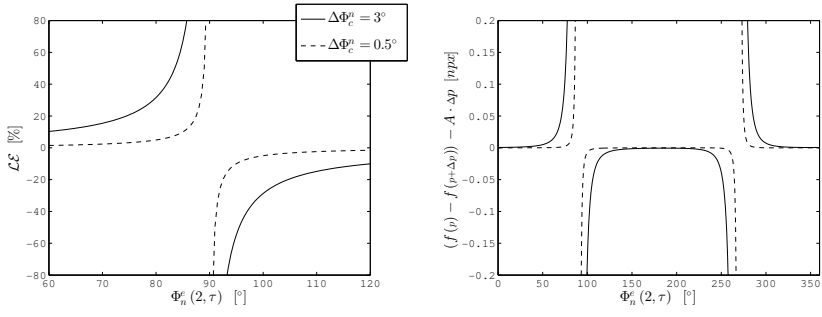


**Figure 5.11.** Linearization error for PD relocalization. Left graph: linearization error $\mathcal{LE}_f(p, \Delta p)$ in percent (see section 3.4.3). Right graph: remainder of the first order Taylor expansion for relocalization $f$ of disparity.

The used parameter trajectory $p_{(\tau)}$ describes a $\Phi^e_{n(2,\tau)}$ camera rotation (along $\mathbf{e}_2$-axis) for the local pose $\Gamma^e_n$ at a fixed position. The landmark is located in front of the initial global pose (state $p_{(0)}$) at a distance of 2 baselines (initial disparity $0.5[npx]$). That is, the global orientation $\Phi^e_{n(2,\tau)}$ determines the angle between the cameras optical axis and the view ray, i. e., for $\Phi^e_{n(2,\tau)} = 90°, 270°$ the landmark intersects the camera plane. The error

$\triangle p$ to be propagated, is chosen as for the ID initialization in section 5.1.4 except for the uncertainty $\Delta\Phi_c^n$ of the relocalization orientation $\Phi_c^n$ (see figure 5.11).

As can be seen, for a variation (uncertainty) of 3° in the local orientation $\Phi_c^n$ the linearization strongly degrades close to the singularities at $\Phi_{c\,(2)}^n = 90°, 270°$. For a view ray at an angle of 60° to the actual camera plane a linearization error of $\sim 10\%$ occurs. Even though the absolute error is $< 0.01[npx]$ for a respective disparity of $> 2[npx]$, this causes a distortion in the covariance prediction. Moving close to the singularities this error increases unbounded. This is similar to the effect of singular initializations in ID (see 5.1.4). Nevertheless, it can occur for out of view landmarks with each pose relocalization, instead solely for the unique landmark initialization. Thus, landmarks have to be excluded from the system state, when they leave the cameras field of vision, come close to it's camera plane and the relocalization orientation $\Phi_c^n$ is highly uncertain at the same time. This is not the case for ES and ID, since the landmarks are defined globally and thus are independent of the actual cameras pose.

**Local Uncertainty** Note that, in contrast to ID, the local orientation $\Phi_c^n$ causes this linearization error. That is, the orientation uncertainty in relocalization is the actual prediction's uncertainty instead the one for the global orientations. By this, $\Delta\Phi_c^n$ can be assumed to be small depending on the quality of prediction. For low orientation uncertainties, i. e., $< 1°$, the linearization error is negligible for distances of $> 10°$ to the singularities. By this, the amount of the prediction's uncertainty allows for a precise appraisal of valid relocalization areas for out of view landmarks. Thus, in most cases the landmarks can be kept in the filter state for re-matching, providing a low prediction uncertainty between two relocalizations.

**In-View Landmarks** It has to be stated furthermore, that for rotation angles $-45° < \Phi_c^n < 45°$ the linearization errors are insignificant (see right graph in figure 5.11). This corresponds to view rays having an absolute angle to the cameras optical axis of less than 45°. Since typical stereo cameras do not have a view angle of more than 90°, the relocalizations linearization error for all active (in-view) landmarks is negligible.

## 5.5 Conclusion

Concluding the model evaluation, results in contrary advantages and drawbacks for the three landmark representations. The most obvious differences are the linearization and PDF representation errors in dependence on pose uncertainties.

The ES representations allows for a consistent and informative initialization of close landmarks. This holds for poses with moderate uncertainty or high position uncertainty. In contradiction to the analysis made by Paz et al. in [Paz+08], using highly uncertain orientations for the initial poses results in significant inconsistencies in the predicted PDF. Similar effects were observed by Sünderhauf et al. in [SLP07] and by Bailey et al. in [Bai+06]. By this, even for small camera to landmark distances ($\geqslant 1$ $[b]$) the ESs initialization process is not capable of covering the true error propagation mapping. This results in an invalid representation of the landmark's position, and by this, it's correlations with the systems pose. Increasing the landmark's distance to the camera leaves the ESs predicted PDF inconsistent and significantly non informative ($\mathcal{I} \ll 0$). Moreover, the observation model cannot cover the landmark's PDF deformation during projection. Due to the inconsistent covariance propagation during projection the KF will give the predicted observations a too high weight.

The ID landmark models are consistent and informative for moderate orientation uncertainties $\leqslant 3°$ in all configurations. In contrast to ES, it's initialization model fully covers the true error propagation for all camera to landmark distances. The only limitations are the poses orientation uncertainty and the singularity in $\phi$ and $\psi$ respectively. Due to the neglected second order derivative for the view ray initialization standard deviations of $\geqslant 5°$ cause significant inconsistencies. In combination with view rays, that are almost parallel to the up axis ($\mathsf{e}_2$), high errors in the initialized covariance occur. Moreover, the high linearization error for high changes in view directions reduces the IDs performance. Since the reprojections true PDF deviates highly from a Gaußian distribution, this drawback is system inherent. Thus, it cannot be solved assuming Gaußian distributions for observation predictions, i. e., landmark projections in Kalman Filters.

The PD parametrization and the respective initialization model are able to cover the true error propagations for all tests. Especially due

to the strictly linear initialization consistency of the system models is ensured even for highly uncertain poses. Except for the large changes in the observations view directions this also holds for the observation models. Even though the observations linearization error is reduced with respect to ID, the relocalization process has to be accepted. This process requires only little computational effort, but complicates the management of inactive landmarks. Moreover, it prohibits the propagation of inactive landmarks, that are close to the camera plane. Nevertheless, this relocalization allows for the local parametrization, offering several advantages. These are the simple adjustment constraints and the incremental estimation, being independent of global drift. The strongest constraint to PD landmarks is, that they are restricted to stereo vision systems.

These results fit the analysis of ID initializations made by Montiel et al. in [CDM08]. They used a similar linearity measure, to evaluate the initialization phase of a landmark. In especially they analyzed the amount of linearization errors for high uncertainties in the inverse depth $\rho$. They concluded, that a direct connection between the uncertainty of $\rho$, the amount of change in view angle and the linearization error exists. Even though the depth initialization for stereo systems is much better compared to monocular systems, this effect still has a significant impact to the landmarks observation. For PD the same linearization error results, when the relocalization frequency is too low. It can be concluded, that the linearization error in PD is due to the same modeling error as in ID. That is, parts of the conclusions drawn by Montiel et al. for IDs observation prediction, can be transferred to the disparity in the PD representation.

Chapter 6

# Filter Implementation

This chapter describes the work flow and implementation issues of the proposed software system. After initializing the filter state, the system is executed in a loop applying the image processing, the state prediction and the SLAM estimation. The initialization provides the filter state and the respective covariances for the filter startup. The data processing captures the sensor data and the images, and applies feature tracking. In the prediction the acquired IMU data is used to predict the navigation state. Following that, an update preliminary is performed, to prepare the data structures for the filter update. Finally, the estimation step applies the Kalman Update and some post processing on the data structures. For the PD representation the relocalization is done in addition. The complete work flow is schematically visualized in figure 6.1. The software framework used for IMU to camera calibration is described in the final section.

## 6.1 Data Structures

A fundamental part of the filter implementation is the choice of appropriate data structures. Beside the filters state vector and covariance matrix additional information has to be available. On the one hand, information on the landmarks visual appearance have to be recorded for visual tracking. On the other hand, parts of the system state not included in the filter state have to be accessible.

To allow for tracking landmarks over time, descriptors for the respective visual features are needed. Therefore, the SIFT descriptors (see section 2.5) for all active landmarks are stored. For the representation of the incremental rotations $\Phi = \{R, \phi\}$ only $\phi$ is included in the state vector and it's covariance matrix. The rotation matrix $R$ is stored externally and is updated every time

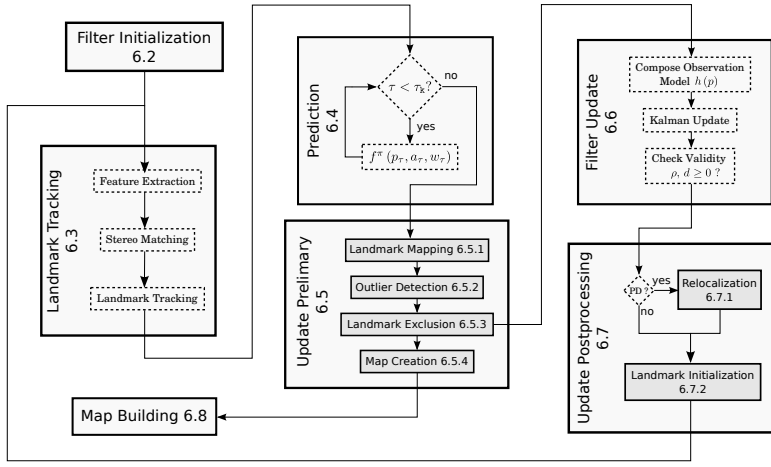


**Figure 6.1**. Visualization of work flow for the proposed SLAM system.

the incremental rotation is normalized (see section 3.3.4). This is applied after each prediction step (see section 6.4) and in the update post-processing (see 6.7). Thus the full filter state structure $\mathcal{S}_{\mathtt{k}}$ for time $\mathtt{k}$ is

$$\mathcal{S}_{\mathtt{k}} = \left\{ p_{\mathtt{k}},\, C_{p_{\mathtt{k}}},\, R_i^n,\, R_n^e,\, R_i^c,\, S^1 \cdots S^{\mathtt{N}} \right\} \tag{6.1.1}$$

Here, $p_{\mathtt{k}}$ and $C_{p_{\mathtt{k}}}$ are the actual filter state and it's covariance respectively. The rotation matrices $R_i^n$, $R_n^e$ and $R_i^c$ are corresponding to $\Phi_i^n$, $\Phi_n^e$ and $\Phi_i^c$ after the preceeding normalization. Finally, $S^{\mathtt{i}} = \left(S_x^{\mathtt{i}}, S_y^{\mathtt{i}}\right)$ represent the SIFT descriptors from the master and slave image respectively, that have been lastly matched with the landmark.

The captured sensor data is stored in a structure $\mathcal{D}$ containing the raw data and the respective time stamps. Each is made up of the captured master-slave image pair $I_{\mathtt{k}}^c, I_{\mathtt{k}}^{c'}$ for time $\tau_{\mathtt{k}}$ and all IMU samples $s_{\mathtt{i}}$ with timestamps $\tau_{k-1} < \tau_{s_{\mathtt{i}}} \leqslant \tau_{\mathtt{k}}$. In addition to that, all detected (see section 6.3) visual features $F^1 \ldots F^{\mathtt{M}}$ are stored. Beside their SIFT descriptors, $F^{\mathtt{i}}$ includes the matched stereo correspondences $x_{\mathtt{i}} \leftrightarrow y_{\mathtt{i}}$. A set $\mathcal{C}$ describes the successfully matched landmarks and the respective stereo correspondences. That is,

$(\mathtt{i}, \mathtt{j}) \in \mathcal{C} \;\Leftrightarrow\; S^{\mathtt{i}} \leftrightarrow F^{\mathtt{j}}$.

A special data structure is used for map building. It stores all relevant landmarks and additional information, to allow for improved environment mapping or loop closing. For a detailed description see section 6.8.

## 6.2 Initialization

When the filter is initialized, the initial values for the navigation and nuisance parameters have to be determined. For synthetic tests (see section 7.1, page 153) the ground truth for initialization is known. For initialization in real systems the choice of initial values and uncertainties is crucial. These are strongly dependent on the systems conditions at startup. Thus, they have to be chosen accordant to the assumptions, made for the application area (see chapter 7).

For synthetical tests a prediction without IMU is used. For these tests the first system pose is used as global reference system $e$. Thus, the initialization of the system pose $\Gamma_i^e$ for ES and ID as well as $\Gamma_n^e$ and $\Gamma_i^n$ for PD respectively are known exactly (covariance = 0).

When using IMU control input in real applications initializations for the nuisance (IMU biases etc.) and calibration parameters have to be given additionally. The initial value for the position is chosen to be $t_{ec}^e = 0_3$ with the covariance $0^3$. By this, the position of the global frame $e$ is set to the initial systems position. For the orientation an alignment between the systems $\mathtt{e}_1/\mathtt{e}_3$-plane and the earths north-east-plane has to be performed. That is, $\Phi_i^e$ has to be estimated to determine the gravities orientation (see section 2.2.2). Therefore, it is assumed, that the system is aligned to the gravity ($\Phi_i^e = 1^3$) up to a certain amount of degrees. Depending on the expected deviation the initial orientations standard deviations $\sigma_{\phi(1)}$ and $\sigma_{\phi(3)}$ for $\mathtt{e}_1/\mathtt{e}_3$-rotation are chosen. Since the heading ($\mathtt{e}_2$-rotation) is fixed to the initials system heading, $\sigma_{\phi(2)}$ is set to 0. The values and standard deviations $\sigma_{v_0}$ for the velocity are chosen depending on the expectations of the systems initial conditions. According to the navigation state $\pi_0$ (see definition (4.2.8), page 77) the initial state covariance $C_{\pi_0}$ containing no

landmarks is:

$$\text{global navigation:} \quad C_{\pi_0^g} = diag\left(0^3, \sigma_{v_0}^2 \cdot 1^3, \sigma_{\phi_x}^2, 0, \sigma_{\phi_z}^2\right) \tag{6.2.1}$$

$$\text{local navigation:} \quad C_{\pi_0^l} = diag\left(0^3, \sigma_{v_0}^2 \cdot 1^3, 0^3, \sigma_{\phi_x}^2, 0, \sigma_{\phi_z}^2\right) \tag{6.2.2}$$

The standard deviations for the IMUs nuisance parameters and the IMU-camera calibration are given by the sensors hardware specifications and the determined camera-IMU calibration (covariance $C_{ci}$, see section 6.9) respectively. The standard deviations for the IMU biases and the scales are denoted as $\sigma_{b_a}, \sigma_{b_w}$ and $\sigma_{s_a}, \sigma_{s_w}$ respectively. By this, the initial covariance for $\eta_0$ is:

$$C_{\eta_0} = diag\left(\sigma_{b_a}^2 \cdot 1^3, \sigma_{b_w}^2 \cdot 1^3, \sigma_{s_a}^2, \sigma_{s_w}^2, C_{ci}\right) \tag{6.2.3}$$

$$\tag{6.2.4}$$

For systems, not modeling the IMU-camera calibration, the matrix $C_{ci}$ is omitted. Finally, the initials system covariance is:

$$C_{p_0} = \begin{pmatrix} C_{\pi_0} & 0 \\ 0 & C_{\eta_0} \end{pmatrix} \tag{6.2.5}$$

## 6.3 Landmark Tracking

The landmark tracking is performed visually using the SIFT algorithm, as described in section 2.5. For the master $I_{\mathtt{k}}^{c}$ and slave $I_{\mathtt{k}}^{c'}$ images, captured at the actual time $\tau_{\mathtt{k}}$, all available SIFT features are extracted. Since this is independent of the remaining work flow, it is performed in a separate thread in parallel. This allows for minimization of system runtime, because the feature extraction is the most time expensive processing step.

After detecting all possible features the matching between $I_{\mathtt{k}}^{c}$ and $I_{\mathtt{k}}^{c'}$ is performed. That is, all features $x$ and $y$ in the master and slave camera respectively are matched. This results in a set of stereo correspondences $x_{\mathtt{j}} \leftrightarrow y_{\mathtt{j}}$. To relate those to the landmarks in the state structure $\mathcal{S}$, a second SIFT matching is performed between these correspondences and the stored $S^{\mathtt{i}}$ descriptors. This results in matches $\mathtt{i} \leftrightarrow \mathtt{j}$ between the landmarks and the observed features. Afterwards, the tuples $(\mathtt{i}, \mathtt{j})$ are composed to the set

$\mathcal{C}$ for the senor data structure $\mathcal{D}$.

**Matching Constraints** During the stereo matching certain constraints are applied. That is, stereo correspondences are known to have a similar height and a disparity $> 0$. Since both constraints apply to the normalized pixels, a disparity close to 0 marks a correspondence distance of $\ll 1$ pixel. Such small disparities, i. e., below the camera resolution, represent points being very far from the camera. Keeping in mind the possible occurrence of points behind the camera (see section 4.2.1, page 73), this constraint supports the estimators validity. It prevents initializations of landmarks behind the camera, or following observations indicating such points.

## 6.4 Prediction

To apply the prediction models given in section 4.2.5, an iterated prediction has to be used. This is due to the fact, that the measurements given by the IMU are captured with a sample-rate much higher than for the cameras. To compensate for this, the time difference $\tau_\mathtt{k} - \tau_{\mathtt{k}-1}$ between two image measurements is partitioned into time steps corresponding to the IMU timestamps.

Although the sample rates for both systems are fixed, the time steps for the IMU samples cannot be considered as constant. Because the cameras frame-rate is no integral fraction of the IMUs sample rate in general, an image might be captured at time $\tau_\mathtt{k}$ between two succeeding IMU samples for time $\tau_\mathtt{k} - \epsilon_1$ and $\tau_\mathtt{k} + \epsilon_2$. In this case, the sample is divided into two samples (see figure 6.2). They have the same acceleration and turn rate

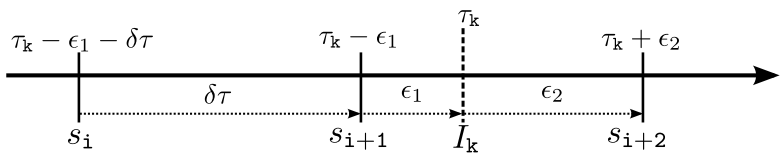


**Figure 6.2.** Partitioning an IMU sample for consistent time intervals for IMU samples $s$ and images $I$.

data, but the time interval length is divided from $\delta\tau = \epsilon_1 + \epsilon_2$ to $\epsilon_1$ and $\epsilon_2$. By this, the movement before and after the image can be approximated using portions of the according IMU sample.

For each prediction step the models (4.2.20) and (4.2.21) for global and local navigation respectively (see page 80) are applied using the determined time difference. Following that, the state covariance is propagated in time using the prediction's Jacobian (see equation (3.3.6) on page 47). Finally the prediction's uncertainty is added to the propagated state covariance. For this the given process noise covariance $C_\omega$ and the error propagation matrix $W_\mathtt{k}$, determined as in section 4.3, are used. Since the choice of the process noise $C_\omega$ is crucial for parameter tuning in Kalman Filtering, it is discussed further in chapter 7 for the respective hardware setups.

After each prediction step the predicted orientations $\Phi_i^e$ and $\Phi_i^n$ for global and local navigation respectively are normalized. This is done as described in section 2.2.1 on page 13. Additionally, according to section 3.3.4 the respective error propagation is applied to the predicted state covariance.

## 6.5 Update Preliminary

For updating the filter state the detected stereo correspondences $(x_1 \leftrightarrow y_1)$ $\ldots$ $(x_\mathtt{M} \leftrightarrow y_\mathtt{M})$ (see above) are related to the filters landmarks $\mathcal{X}_1$ $\ldots$ $\mathcal{X}_\mathtt{N}$. Therefore, some mappings are defined, to classify landmarks as matched, lost etc. Since the state and observation models use normalized pixel coordinates, the features and their covariances have to be transformed accordingly. This is done using the methods described in section 2.4.1, page 28. After the landmark matching an outlier detection is applied and invalid landmarks are removed from the system state.

### 6.5.1 Landmark Mapping

To be able to apply the Kalman Update, the landmarks $\mathcal{X}_\mathtt{i}, \mathtt{i} \in \{1 \ldots \mathtt{N}\}$ and the determined stereo correspondences $\binom{x_\mathtt{j}}{y_\mathtt{j}}, \mathtt{j} \in \{1 \ldots \mathtt{M}\}$ have to be related. This is accomplished using the mapping $\mathcal{C}$ between the landmarks and the stereo correspondences determined in section 6.3.

Using the set $\mathcal{C}$ of correspondences two match mappings can be defined,

that describe the matching state of landmarks and observations. Let

$$m^{\mathcal{X}}_{xy(\mathtt{j})} := \begin{cases} \mathtt{i} & \text{if} \quad \exists\, \mathtt{i} : (\mathtt{i}, \mathtt{j}) \in \mathcal{C} \\ 0 & \text{else} \end{cases} \tag{6.5.1}$$

$$m^{xy}_{\mathcal{X}(\mathtt{i})} := \begin{cases} \mathtt{j} & \text{if} \quad \exists\, \mathtt{j} : (\mathtt{i}, \mathtt{j}) \in \mathcal{C} \\ 0 & \text{else} \end{cases} \tag{6.5.2}$$

These mappings define the connections between the landmarks indices in the filter state and the indices of detected stereo correspondences in the measurement vector $\left(\begin{smallmatrix} x_1 & y_1 & \cdots & x_{\mathtt{M}} & y_{\mathtt{M}} \end{smallmatrix}\right)$. Note that the mappings are not inverse, since none of them is injective in general. On the one hand, (6.5.1) maps all newly detected stereo correspondences (i. e., new landmarks) to 0. On the other hand, (6.5.2) maps the lost landmarks to 0. For the following update procedure let

$$\mathcal{M}_L := \{\, \mathtt{i} \in \{1 \ldots \mathtt{N}\} \mid m^{xy}_{\mathcal{X}(\mathtt{i})} = 0 \,\} \tag{6.5.3}$$

the set of lost landmarks for the actual filter state.

### 6.5.2 Outlier Detection

The detection of outliers is done utilizing the method discussed in section 3.4.1. In the following let $\mathtt{j}$ reference an active stereo correspondence, i. e., $m^{\mathcal{X}}_{xy(\mathtt{j})} \neq 0$. By using the stereo correspondences $\left(\begin{smallmatrix} x_{\mathtt{j}} \\ y_{\mathtt{j}} \end{smallmatrix}\right)$ and the observation model $h(\pi, \mathcal{X})$ defined in equation (4.2.22) an outlier constraint can be defined. Let

$$z_{\mathtt{j}} := \left(\begin{smallmatrix} x_{\mathtt{j}} \\ y_{\mathtt{j}} \end{smallmatrix}\right) - h\left(\pi, \mathcal{X}_{m^{\mathcal{X}}_{xy}(\mathtt{j})}\right) \tag{6.5.4}$$

$$H_{\mathtt{j}} := \frac{\partial h\left(\pi, \mathcal{X}_{m^{\mathcal{X}}_{xy}(\mathtt{j})}\right)}{\partial p} \tag{6.5.5}$$

As shown in section 3.3, $z_{\mathtt{j}}$ is a sample from a zero mean random variable with covariance $H_{\mathtt{j}} C_p H_{\mathtt{j}}^T + C_{xy}$. Given a probability $\mathtt{p}$ let $\epsilon_{\mathtt{p}}$ the limit determined using the inverse $\chi^2_4$ mapping on the probability $1 - \mathtt{p}$, as described in

section 3.4.1. By this, observation $\mathtt{j}$ is an inlier with a probability of $\mathtt{p}$ if and only if

$$\epsilon_{\mathtt{p}} \quad \geqslant \quad z_{\mathtt{j}}^T \cdot \left(H_{\mathtt{j}} C_p H_{\mathtt{j}}^T + C_{xy}\right)^{-1} \cdot z_{\mathtt{j}} \tag{6.5.6}$$

This test is applied to all landmark matches $\mathtt{j} \leftrightarrow m_{xy(\mathtt{j})}^{\mathcal{X}}$. For the following update procedure let

$$\mathcal{M}_{\mathtt{p}} \quad := \quad \left\{ \mathtt{j} \in \{1 \ldots \mathtt{M}\} \;\middle|\; \epsilon_{\mathtt{p}} \;<\; z_{\mathtt{j}}^T \cdot \left(H_{\mathtt{j}} C_p H_{\mathtt{j}}^T + C_{xy}\right)^{-1} \cdot z_{\mathtt{j}} \right\} \tag{6.5.7}$$

the set of outliers in the actual observation vector. Since the true outlier rate is not known in general, $\mathtt{p}$ has to be chosen heuristically.

In the tests applied in chapter 7, a value of $\mathtt{p} = 0.75$ was used. For the camera hardware and feature tracking method used in this thesis, this value has proven to be an appropriate setting. Providing a more stable feature tracking method the value can be increased to increase the number of available correspondences. On the other hand, a less stable tracker implies a higher rejection rate. Since this entails the danger of filter divergency due to complete observation rejections, it has to be done carefully. Thus, the rejection rate has to be chosen in combination with the noise model tuning (see section 4.3).

### 6.5.3 Landmark Exclusion

Some landmarks can be lost in the master or slave camera respectively, due to failed visual matching $\mathcal{M}_L$ or outlier rejection for their observations $\mathcal{M}_{\mathtt{p}}$. These landmarks have been determined above. Moreover, matched landmarks might be rejected, because a limit $\mathtt{l}$ for the maximum number of landmarks, that are used for update, has been exceeded. This limit is needed, since the computational effort for the Kalman Update strongly depends on the state size. Thus, the usage of too many landmarks results in a violation of the real-time constraints. See section 7.1.1 on page 162 for an evaluation of different limit settings. Let

$$\mathcal{M}_{\mathtt{l}} \subseteq \{1 \ldots \mathtt{N}\} \quad : \quad \left|\{1 \ldots \mathtt{N}\} \setminus \left(\mathcal{M}_L \;\cup\; m_{xy(\mathcal{M}_{\mathtt{p}})}^{\mathcal{X}} \;\cup\; \mathcal{M}_{\mathtt{l}}\right)\right| \leqslant \mathtt{l} \tag{6.5.8}$$

the set of landmark indices that have been rejected due to the limit $\mathtt{l}$. Note that the lost and rejected landmarks are not included in the limit test. By

this, the set $\mathcal{M}_R$ of indices for landmarks, that have to be removed from the filter state, is:

$$\mathcal{M}_R \quad = \quad \mathcal{M}_L \ \cup \ m_{xy}^{\mathcal{X}}(\mathcal{M}_{\mathtt{p}}) \ \cup \ \mathcal{M}_{\mathtt{l}} \tag{6.5.9}$$

Removing landmarks is done, by cutting the respective parameters out of the system state. Accordingly, the rows and columns representing the landmarks covariance and correlations with the filter state are removed from the systems covariance. For the filter state $p$ the landmark $\mathtt{i}$ is removed using:

$$p \quad = \quad \begin{pmatrix} \pi \\ \eta \\ \mathcal{X}^{1..\mathtt{i}-1} \\ \mathcal{X}^{\mathtt{i}} \\ \mathcal{X}^{\mathtt{i}+1..\mathtt{N}} \end{pmatrix} =: \begin{pmatrix} p_1 \\ \mathcal{X}^{\mathtt{i}} \\ p_2 \end{pmatrix} \mapsto \begin{pmatrix} \pi \\ \eta \\ \mathcal{X}^{1..\mathtt{i}-1} \\ \mathcal{X}^{\mathtt{i}+1...\mathtt{N}} \end{pmatrix} = \begin{pmatrix} p_1 \\ p_2 \end{pmatrix} \tag{6.5.10}$$

$$C_p \quad = \quad \begin{pmatrix} C_{p_1} & C_{p_1,\mathcal{X}^{\mathtt{i}}} & C_{p_1,p_2} \\ C_{\mathcal{X}^{\mathtt{i}},p_1} & C_{\mathcal{X}^{\mathtt{i}}} & C_{\mathcal{X}^{\mathtt{i}},p_2} \\ C_{p_2,p_1} & C_{p_2,\mathcal{X}^{\mathtt{i}}} & C_{p_2} \end{pmatrix} \mapsto \begin{pmatrix} C_{p_1} & C_{p_1,p_2} \\ C_{p_2,p_1} & C_{p_2} \end{pmatrix} \tag{6.5.11}$$

where (6.5.11) results from linear error propagation. To reduce the computational effort, this is applied to all landmarks $\mathtt{i} \in \mathcal{M}_R$ at once.

Note that all landmarks, that have been lost in a single (master or slave) image are removed from the state, although an update using only one observation ($x$ or $y$) would be possible. Since such observations provide significantly reduced information on the system state (2 instead of 4 observation components), the landmark slots are freed for more informative ones.

Moreover, all landmarks, that have been lost in a single time step, are removed from the state, although they could be held in the filter for a possible rematch. This is done, because the limited number of landmark slots is reserved for landmarks, that can be used in the actual images. Although the computational effort for the Kalman Update is only slightly increased by non observed landmarks, the introduction of new landmarks is effected directly. In contrast to ES and ID this is no issue to be considered

for the PD representation (see evaluation in section 7.2.2, page 187). But since PD landmarks are parametrized in the local camera frame $c$, it is hard to determine, which landmarks are worth, to be kept in filter state.

Designing methods for optimally exploiting landmarks, that have been lost in a single image or for limited number of images, is subject to further development. The most difficult task to solve, is the selection of landmarks being worth the temporally "wasted" landmark slots.

### 6.5.4 Map Creation

During the process of landmark exclusion landmarks are selected, that are to be inserted to the landmark database (see section 6.8). To solve this task, the landmark classifications $\mathcal{M}_L$, $m^{\mathcal{X}}_{xy(\mathcal{M}_\mathrm{P})}$ and $\mathcal{M}_1$ are used.

The landmarks $m^{\mathcal{X}}_{xy(\mathcal{M}_\mathrm{P})}$, whose observations have been detected as outliers, are considered as invalid. This is done, because it is not known, whether the observation or the landmarks itself is erroneous. A task of future work is to develop methods, that can be used to determine landmarks, that are likely to be initialized correctly. This can be achieved, for example by analyzing the number of successful landmark observations and the landmarks covariance.

The landmarks to be inserted into the database are those, that are excluded from the filter state and not considered as outliers. That is, the set of landmarks

$$\mathcal{M}_D \quad := \quad \mathcal{M}_R \backslash m^{\mathcal{X}}_{xy(\mathcal{M}_\mathrm{P})} \tag{6.5.12}$$

is added to the database. This is done as described in section 6.8.

## 6.6 Kalman Update

After the landmarks have been matched and, if applicable, removed from the filter state, the Kalman Update can be performed. That is, the observation model $h$ (see section 4.2.6) can be composed. The composed model and the corresponding stereo correspondences are used, to apply the KF update iteration (see section 3.3). In the following, the set

$$\mathcal{M}_U \quad := \quad \{1...\mathtt{N}\} \backslash \mathcal{M}_R \tag{6.6.1}$$

describes the landmarks left in state after landmark exclusion. Note that the indices contained in $\mathcal{M}_U$ still refer to the landmark indices in the filter state before exclusion. Thus, they are used as identifiers not as positions in the state vector.

As already stated, the complete observation model $h(p)$ has to be composed by collecting the sub models $h(\pi, \mathcal{X})$ (see equation (4.2.23), page 81) for all involved landmarks $\mathcal{X}$. To determine the observations, that are connected to the landmarks $\mathcal{M}_U$, the mapping $m_{\mathcal{X}}^{xy}$ from equation (6.5.2) on page 135 is used. The predicted observations are mapped to the stereo observations by:

$$\forall \mathtt{i} \in \mathcal{M}_U : \quad \mathcal{X}_{\mathtt{i}} \leftrightarrow \begin{pmatrix} x_{\mathtt{j}} \\ y_{\mathtt{j}} \end{pmatrix} \iff m_{\mathcal{X}}^{xy}(\mathtt{i}) = \mathtt{j} \tag{6.6.2}$$

This leads to the partial observation model:

$$\forall \mathtt{i} \in \mathcal{M}_U : \quad h(\pi, \mathcal{X}_{\mathtt{i}}) \stackrel{!}{=} \begin{pmatrix} x_{m_{\mathcal{X}}^{xy}(\mathtt{i})} \\ y_{m_{\mathcal{X}}^{xy}(\mathtt{i})} \end{pmatrix} \tag{6.6.3}$$

Subsequent to the composition of $h(p)$ the filter update is performed. The needed tuning parameters for the iterated update termination (maximum update norm and number of iterations, see section 3.3.3) have to be chosen depending on the used models and filter state. I. e., the maximum update norm is dependent on the length of the filter state. Due to this, for the experiments performed in this thesis the updates norm is divided by the square root of the state size. By this, the root mean square norm, that is independent on the length of the state, is checked as termination criterion. As will be shown in section 7.1.3 on page 167, the PD model is the only estimator, that is applicable to the IEKF. Due to inconsistencies in the ES and ID estimations, applying more than one iteration results in a degraded estimation quality.

#### Negative depths and disparities

For ID and PD parametrizations the inverse depth $\rho$ and the disparity $d$ respectively are close to 0 for very distant points. Thus, due to the noisy input data and the used linear estimators, that cannot apply hard

constraints like$\rho, d > 0$, landmarks with negative $\rho$ and $d$ respectively can occur. This effect results in estimations, where landmark positions lie behind the camera. Since such points are inconsistent with the physical imaging process, exception handling has to be applied.

To keep the state valid for such estimations, two alternatives are available. On the one hand, landmarks being inconsistent in that sense can be removed directly from the system state. Therefore, during the outlier detection and the update process all landmarks have to be checked frequently for validity. Due to the feature matching and outlier rejection phases (see sections 6.3 and 6.5.2), the filter state contains only valid landmarks before and after the update operation. Thus, the critical point is the estimation itself. For all applied synthetical and real world tests, using the initialization and observation $\rho, d > 0$, the validity of all landmarks was ensured. For monocular SLAM, the undelayed initialization using a high uncertainty in depth (e. g., see [CDM08], [Sol+12]) bears the risk of points behind the camera during estimation. Since stereo SLAM provides precise landmark initializations and all observations imply $\rho, d \geqslant 0$, this problem has been classified as negligible for ID and PD in stereo SLAM.

## 6.7 Update Post-Processing

After the filter state has been updated using the image measurements, the post processing has to be applied. That is, the relocalization for the PD representation and the initialization of new landmarks are performed. The relocalization is required for the local navigation model, since the navigation frame $n$ is moved to the actual IMU-frame $i$ (see section 4.2.2). Furthermore the local landmarks in the PD representation have to be transferred to the actual camera frame $c$. When the system has been relocalized, the newly detected landmarks can be inserted to the filter state.

Furthermore, as done for the prediction steps, the incremental rotations in the filter state have to be normalized. In connection to this, the filter's state covariance has to be modified (see section 3.3.4, page 51).

### 6.7.1 Relocalization

The relocalization only has to be applied for the local navigation model, i. e., for the PD representation. It is done by concatenating the local and global pose resulting in the new global pose (see section 4.2.7). The new state covariance is computed using linear error propagation with the Jacobian $J_{ni}$, given in appendix A.3.2. This allows for the transition $i \to n$ and $\Gamma_n^i$ to be the new local pose, i.e.:

$$\Gamma_n^i \quad := \quad \left\langle 1^3, 0 \right\rangle \tag{6.7.1}$$

Due to the error propagation from the local to the global pose the new local pose's covariance is $0^6$. This is due to the fact, that the local uncertainty was propagated to the global uncertainty. Thus, the rows and columns of the state covariance $C_p$ corresponding to the local pose are set to 0.

By this, the PDs filter state pose $\pi^l$ is transformed by:

$$\begin{pmatrix} t_{ni}^n \\ v_{ni}^n \\ \Phi_n^i \\ t_{en}^e \\ \Phi_e^n \end{pmatrix} \quad \mapsto \quad \begin{pmatrix} 0_3 \\ \Phi_n^i \cdot v_{ni}^n \\ 1^3 \\ t_{en}^e + \Phi_n^e \cdot t_{ni}^n \\ \Phi_n^i \cdot \Phi_e^n \end{pmatrix} \tag{6.7.2}$$

Additionally all landmarks $\mathcal{X}^{\text{PD}}$ are transformed by (see equations (4.2.46) and (4.2.47), page 86):

$$\mathcal{X}^{\text{PD}} \;=\; \begin{pmatrix} x^{\text{PD}} \\ d \end{pmatrix} \quad \mapsto \quad \begin{pmatrix} \mathbf{H}\left(\mathbf{x}^{\text{PD}}\right) \\ (d \cdot b)^{-1} \cdot P_3 \cdot \Phi_n^c \cdot \left(b \cdot \begin{pmatrix} x^{\text{PD}} \\ 1 \end{pmatrix} - d \cdot t_{nc}^n\right) \end{pmatrix} \tag{6.7.3}$$

The linear error propagation is performed for the landmark and relocalization in combination using the composed Jacobian $J_{ni}$. It is made up of the Jacobian for (6.7.2) and (6.7.3) given in appendix A.3. As stated above, all correlations and variances for the local pose (i. e., $t_{ni}^n$ and $\Phi_n^i$) are 0 afterwards. This is due to the constant mapping of $\Gamma_n^i$, having a Jacobian of 0.

### 6.7.2 Landmark Initialization

After the updated and relocalized filter state has been determined, the new landmarks are to be inserted into the filter state. To solve this task, the models, given in 4.2.4 (page 77), and their Jacobians are used. At first, the stereo correspondences $x_\mathrm{j} \leftrightarrow y_\mathrm{j}$, that have been detected by stereo matching and that are not yet part of the system state (i. e., $m^{\mathcal{X}}_{xy(\mathrm{j})} = 0$), and their covariance $C_{xy}$ are determined. These are used, to apply the initialization models (4.2.9), (4.2.17) and (4.2.18) for ES, ID and PD landmarks respectively. The resulting landmark $\mathcal{X}$ is then appended to the system state $p$ and the according covariance propagation is applied:

$$p \mapsto \begin{pmatrix} p \\ \mathcal{X} \end{pmatrix} \tag{6.7.4}$$

$$C_p \mapsto \begin{pmatrix} C_p & C_p \cdot J_p^T \\ J_p \cdot C_p & J_{xy} \cdot C_{xy} \cdot J_{xy}^T \end{pmatrix} \tag{6.7.5}$$

Where $J_p$ is the derivative of the landmark's initialization with respect to the filter state, i. e., the system pose and camera to IMU calibration. $J_{xy}$ represents the Jacobian for $\mathcal{X}$ by $x$ and $y$.

Since the landmark models use normalized pixels, pixel normalization (see section 2.4.1) is applied before the initialization process. Accordingly, $C_{xy}$ is determined as described in section 4.3. For ID it is possible to introduce landmarks observed only in a single image (master or slave) by assuming a highly uncertain depth, as described in [CDM08]. For PD this can be done in a similar way. This causes biases, when landmarks are mostly less or mostly more distant than the assumed initialization depth. Moreover, the uncertainty area includes infinite distant points, and even worse points behind the camera. Since a main focus of this thesis is on consistency evaluation, this initialization technique is not applied in the system proposed in this thesis.

**For ES and ID** the incorporation of new landmarks is attended by a full covariance propagation. This is due to the fact, that the initialization is dependent on the global pose and the camera to IMU calibration. The Jacobian for the initialization is composed of the sub matrices given in

appendix A.3.1. According to the notation used there and the composition of the filter state (see section 4.2.3, page 76) the Jacobians in (6.7.5) for N active landmarks are

$$J_p^{\text{ES}} = \begin{pmatrix} \frac{\partial \mathcal{X}^{\text{ES}}}{\partial t_{ei}^e} & 0^3 & \frac{\partial \mathcal{X}^{\text{ES}}}{\partial \Phi_e^i} & 0^{3\times 4} & \frac{\partial \mathcal{X}^{\text{ES}}}{\partial \Phi_c^i} & \frac{\partial \mathcal{X}^{\text{ES}}}{\partial t_{ci}^c} & 0^{3\times 3\mathtt{N}} \end{pmatrix} \tag{6.7.6}$$

$$J_{xy}^{\text{ES}} = \begin{pmatrix} \frac{\partial \mathcal{X}^{\text{ES}}}{\partial x} & \frac{\partial \mathcal{X}^{\text{ES}}}{\partial y} \end{pmatrix} \tag{6.7.7}$$

for ES and

$$J_p^{\text{ID}} = \begin{pmatrix} \frac{\partial \mathcal{C}}{\partial t_{ei}^e} & 0^3 & \frac{\partial \mathcal{C}}{\partial \Phi_e^i} & 0^{3\times 4} & \frac{\partial \mathcal{C}}{\Phi_c^i} & \frac{\partial \mathcal{C}}{t_{ci}^c} & 0^{3\times 6\mathtt{N}} \\ 0^{2\times 3} & 0^{2\times 3} & \frac{\partial\,[\,\phi|\,\psi]}{\partial \Phi_e^i} & 0^{2\times 4} & \frac{\partial\,[\,\phi|\,\psi]}{\partial \Phi_c^i} & 0^{2\times 3} & 0^{2\times 6\mathtt{N}} \\ 0 & & & \cdots & & & 0 \end{pmatrix} \tag{6.7.8}$$

$$J_{xy}^{\text{ID}} = \begin{pmatrix} 0^{3\times 2} & 0^{3\times 2} \\ \frac{\partial\,[\,\phi|\,\psi]}{\partial x} & \frac{\partial\,[\,\phi|\,\psi]}{\partial y} \\ \frac{\partial \rho}{\partial x} & \frac{\partial \rho}{\partial x} \end{pmatrix} \tag{6.7.9}$$

for ID. Even though the sparseness of these Jacobians can be exploited, several matrix multiplications have to be applied, to augment the system covariance. Especially for states containing many landmarks, the computational effort increases strongly with increasing state size. Note that $J_p^{\text{ES}}$ has 3 rows and $19 + 3\mathtt{N}$ columns, where N is the number of landmarks in state. In contrast to this, the ID Jacobian $J_p^{\text{ID}}$ has 6 rows and $19 + 6\mathtt{N}$ columns. Keeping in mind the sparseness of these matrices, N matrix multiplications of size $3 \times 19$ times $19 \times 3$ for ES and of size $6 \times 19$ times $19 \times 6$ for ID have to be performed, when inserting a single landmark. Thus, the computational effort for covariance propagation is increased by a factor of 4 for ID, compared to ES.

**For PD** in contrast to ES and ID, the initialization of landmarks is independent of the system pose (see model (4.2.18), page 79). Thus, the Jacobian $J_p^{\text{PD}}$ vanishes. Moreover, the $J_{xy}^{\text{PD}}$ is constant (see model (4.2.19)):

$$J_{xy}^{\text{PD}} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 & \frac{1}{2} \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{6.7.10}$$

Thus, the covariance propagation (6.7.5) simplifies to:

$$C_{p'} = \begin{pmatrix} 1\!\!1 & 0 \\ 0 & J^{\text{PD}} \end{pmatrix} \cdot \begin{pmatrix} C_p & 0 \\ 0 & C_{xy} \end{pmatrix} \cdot \begin{pmatrix} 1\!\!1 & 0 \\ 0 & J^{\text{PD}} \end{pmatrix}^T \tag{6.7.11}$$

$$= \begin{pmatrix} C_p & 0 \\ 0 & J_{xy}^{\text{PD}} \cdot C_{xy} \cdot J_{xy}^{\text{PD}\,T} \end{pmatrix} \tag{6.7.12}$$

As can be seen, in contrast to ES and ID the augmented state covariance simply arises from augmenting $C_p$ by $J^{\text{PD}} C_{xy} J^{\text{PD}\,T}$ on the lower right. That is, no complete covariance transformation has to be applied, but only a small sub matrix has to be appended. Moreover, using a constant transformation for normalization of pixels (see section 2.4.1), $J_{xy}^{\text{PD}} \cdot C_{xy} \cdot J_{xy}^{\text{PD}\,T}$ is constant and can be computed in advance.

**Landmark Limit** Due to the increased complexity for ES and ID landmark initializations, the computational effort of inserting landmarks to the filter state strongly increases with the number of landmarks. This holds for the landmarks, that are already part of the filter state, as well as for the new landmarks. Thus, a limit for the number of initialized landmarks is given. For the PD representation the simple initialization methods allows for inserting all new stereo correspondences, resulting in a more dense map reconstruction. For an evaluation of the landmark initializations computational effort see chapter 7 (pages 187ff).

## 6.8 Map Building

Beside pose estimation, map building is an essential part of SLAM systems. It enables the system to plan routes and detect obstacles in the scouted environment. Moreover, it provides the opportunity to recognize places, that have already been visited. By this, drifts in the pose estimation can be compensated by applying adjustment techniques (see section 4.2.8).

The most simple environment maps are made up of point clouds as a collection of detected landmarks. Often a database of descriptors for visual features (e. g., SIFT, see section 2.5) corresponding to these landmarks is created additionally. Therefore, key frames are selected and the extracted feature descriptors along with the according landmarks are stored. This database can be used for adjustment as mentioned above. For advanced mapping, dense 3D-models can be reconstructed, giving precise information on the surrounding area. This is highly time consuming for vision only SLAM systems. Thus, it's use is mostly restricted to active sensing devices, such as radar, sonar, laser scanners and the like. Since the landmarks reconstruction in the systems discussed in this thesis is performed by image processing exclusively, only point clouds are considered here. These are build up by gathering all information on landmarks, that have been used during the estimation process (see section 6.5.4).

### 6.8.1 Database Structure

Landmarks are collected over time storing the landmarks final estimation, i. e., the last filter it is used in. To do so, the landmark is extracted from the filter state, at the time step it was observed lastly. Landmarks detected as outliers are discarded and not added to the map. To improve the environment map created this way, certain constraints can be used, to exclude uncertain or error prone landmarks. I. e., landmarks with a high uncertainty (observed only in a single image) can be omitted for visualization, since they tend to be imprecise. Moreover, landmarks that are afar from the observing cameras have a high uncertainty in the point-camera distance. Thus, they would disturb the visual appearance of the environment map. For loop closing techniques these landmarks can be recorded nevertheless, since they still provide information on the estimated trajectory. The landmarks, afar from

the observing cameras for example, deliver rich information on the systems orientation.

The database used in the proposed SLAM system consists of all landmarks, that have been part of the filter state a least once. Only the outliers, that have been detected as described in the update preliminary section 6.5, are discarded. The landmarks, introduced to the filter state and rejected due to the limitation of the state size, are also included in the database. Although these are very uncertain, they can be used for probabilistic map modeling (e. g., see [GFP08]).

For each landmark $\mathcal{X}$ a descriptor $\mathcal{L}$ is inserted to the database. It consists of a timestamp $\mathtt{k}$, a global pose $\Gamma_e^c$, the landmarks parametrization $\mathcal{X}$, their covariance $C_{\Gamma_e^c,\mathcal{X}}$ as well as the corresponding 3-space point $t_{e\mathcal{X}}^e$. Additionally the landmarks SIFT descriptors $S$ are recorded. That is:

$$\mathcal{L} \;=\; \left( \mathtt{k} \;,\; \Gamma_e^c \;,\; \mathcal{X} \;,\; C_{\Gamma_e^c,\mathcal{X}} \;,\; t_{e\mathcal{X}}^e \;,\; S \right) \tag{6.8.1}$$

The timestamp, pose, landmark and covariance are chosen, to be the ones from the state, the landmark was extracted from. The extraction is done in combination with the landmark exclusion, as is described on page 138.

The covariance $C_{\Gamma_e^c,\mathcal{X}}$ stores the uncertainties of the pose and the landmark as well as their correlations. Since $\Gamma_e^c = \Gamma_i^c \circ \Gamma_e^i$ is made up of the global system pose, the local pose if applicable and the camera calibration, these have to be concatenated. Accordingly, the covariance $C_{\Gamma_e^c,\mathcal{X}}$ has to be computed by linear error propagation from the systems covariance for time $\tau_{\mathtt{k}}$. The 3-space point is computed, as described in the following section. When the landmark is matched to a stereo correspondence $x \leftrightarrow y$, the according SIFT descriptors $S = (S_x, S_y)$ are related to the landmark $\mathcal{X}$. Thus, when a landmark is stored in the database, $S$ contains the descriptors of the latest landmark match.

### 6.8.2 Point Cloud Reconstruction

Beside the landmark database a point cloud of Euclidean 3-space points is created. It can be used, to visualize the estimated system pose and environment map as a simplified 3D-model. For the proposed SLAM system this model is visualized in a user interface, to allow for a first visual analysis

of the estimation process. To improve the visual appearance, the point cloud can be colored using the image features matched with the corresponding landmark. In the following, the extraction of the 3-space points from the landmark descriptor $\mathcal{L}$ is given.

**ES** parametrizations encode the Euclidean 3-space point. Thus, their parameters $\mathcal{X}^{\text{ES}}$ can be used directly as representation for the reconstructed landmark.

**ID** parametrizations encode the 3-space point implicitly, using the hookup point, the view ray and the inverse depth $\mathcal{X}^{\text{ID}} = (\, C \;\phi\;\psi\;\rho\,)$. These have to be used, to build up the corresponding map point $X$. This is accomplished by:

$$X \;=\; C + \frac{1}{\rho} \cdot \begin{pmatrix} \sin(\phi) \cdot \cos(\psi) \\ -\sin(\psi) \\ \cos(\phi) \cdot \cos(\psi) \end{pmatrix} \tag{6.8.2}$$

**PD** Since the PD representation $\mathcal{X}^{\text{PD}} = (\, x^{\text{PD}} \; d\,)$ models the landmark relative to the actual pose $\Gamma_e^c$, the 3-space point $X$ has to be computed by unprojection. To do so, at first $X$ is reconstructed in the local camera frame using the stereo baseline $b$:

$$t_{cX}^c \;=\; \frac{b}{d} \cdot \begin{pmatrix} x^{\text{PD}} \\ 1 \end{pmatrix} \tag{6.8.3}$$

The second step is, to transform the local landmark $t_{cX}^c$ to the global frame using $\Gamma_e^c = \langle \Phi_e^c, t_{ec}^e \rangle$:

$$X \;=\; t_{eX}^e \;=\; \Gamma_c^e\,(t_{cX}^c) \;=\; \Gamma_e^{c\,-1}\,(t_{cX}^c) \tag{6.8.4}$$

$$\;=\; \Phi_e^{cT} \cdot t_{cX}^c + t_{ec}^e \tag{6.8.5}$$

$$\;=\; \frac{b}{d} \cdot \Phi_e^{cT} \cdot \begin{pmatrix} x^{\text{PD}} \\ 1 \end{pmatrix} + t_{ec}^e \tag{6.8.6}$$

## 6.9 IMU-Camera Calibration

When fusing information from camera and IMU measurements, it is important, to consider the coordinate transformation $\Gamma_c^i$ (see figure 4.2). Since both deliver data in their local coordinate system, a common frame has to be used as reference. Therefore, the coordinate transformation $\Gamma_c^i$ between camera $c$ and IMU $i$ frame has to be determined. Although it is possible to model and estimate this transformation within the SLAM process (see appendix A.2.2, page 219), the estimations convergence radius is limited. To overcome this constraint, an initialization for the alignment is estimated in advance. This is done using the software proposed earlier by the author in [PK12b].

Using this method, it is possible to estimate even large misalignments and to determine the covariance matrix for the estimated calibration. These are used, to initialize the calibration parameters ($\Phi_c^i$ and $t_{ci}^c$ in section 4.2.3) and the respective part of the systems covariance matrix. The calibration is performed in a two-stage online process, using a marker pattern. The marker pattern (see figure 6.3) is made up of 5 2D-markers, arranged to a cube with known alignment. Thanks to the capability of online estimation, $\Gamma_c^i$ can be estimated within a few minutes in total. Other methods have been developed (see [LD07], [MR08]), that are applied offline and are expected to result in higher estimation quality. The drawback of such an offline process is, that the used trajectory cannot be chosen at estimation time. I. e., when a trajectory was recorded, that delivers inadequate estimation results, a new trajectory has to be created. Other online methods (see [KS09], [HSG08]) suffer from limitations to magnitudes of the misalignment allowed, to ensure a stable estimate. Especially the orientation deviation is limited to about a few degrees.

The basic idea behind the algorithm is, to estimate a guess for the rotational alignment, in advance to fully estimate the calibration. Therefore, a two-stage IEKF has been designed. It makes use of two different prediction models and state parametrizations. In both parametrizations the navigation frame $n$ is chosen to be the frame of the used marker pattern. Since the heading of the $n$ frame with respect to the $e$ frame is given by the markers orientation, only two angles are used to determine $\Gamma_e^n$ (see relations between $\varepsilon$ and $e$ in section 2.2.2, page 14). Determining $\Gamma_e^n$ is needed, since the

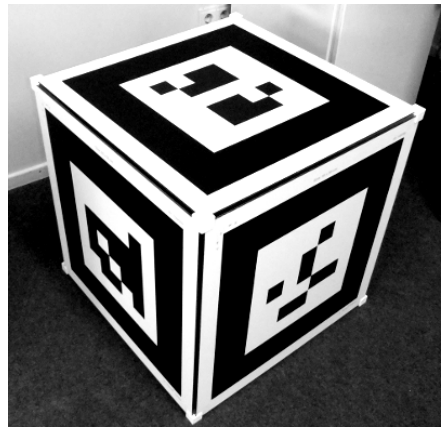

**Figure 6.3.** Marker cube used for IMU-camera calibration. Size: $60 \times 60 \times 60$ [cm]; Marker points $5 \times 8 = 20$

earths gravitation cannot be assumed to be aligned with the marker frame $n$.

#### First Stage Calibration

For the initial estimation of the rotational alignment, the system pose is represented by the cameras coordinate transformation $\Gamma_n^c$. Beside this pose, the gravitational alignment $\Phi_e^n$, the velocity $v_{nc}^n$ and the IMUs nuisance parameters $\eta$ (see definition (4.2.7), page 77) are included in the filter state $p$. Finally the IMU-camera calibration $\Gamma_c^i$ is also included. Thus:

$$p = \begin{pmatrix} t_{nc}^n & v_{nc}^n & \Phi_n^c & \Phi_e^n & \eta & t_{ci}^c & \Phi_c^i \end{pmatrix} \tag{6.9.1}$$

The landmarks (correspondences on marker cube) are not included in the system state, since they are known exactly in the marker frame $n$.

The position prediction is done using a constant velocity assumption, i. e., no acceleration measurements are used. For the orientation prediction the IMUs turn rate measurements are used. Therefore, the angle increments are rotated to the camera frame, using the orientational alignment $\Phi_c^i$ estimated

so far. Thus, the prediction $f(p)$ of the systems position, velocity and orientation is:

$$f(p) = \begin{pmatrix} t_{nc}^n + \Delta T \cdot v_{nc}^n \\ v_{nc}^n \\ \Phi_n^c + \Delta T \cdot \Phi_i^c \cdot (s_w \cdot w + b_w) \end{pmatrix} \tag{6.9.2}$$

After capturing an image and detecting the marker, the resulting 2D/3D correspondences are used for the Kalman Update. To do so, the ES observation model (4.2.24) from section 4.2.6 (replacing frame $e$ by $n$) in combination with the known 3D points is used. Note that the IMU-camera alignment $\Phi_c^i$ is used for prediction only. By this, the calibration is decoupled from the update equation, and is estimated implicitly. Therefore, the updates of the respective state components are kept small, allowing for a precise linearization of the used incremental rotations.

After the estimated standard deviations for the rotational alignment fall beneath a certain threshold, the system switches to the second stage. The threshold is chosen to be 1°, since this restricts following updates to be not larger then 3°. This keeps the rotation linearization stable during the second stage. Switching to the second stage is done, by transferring the state to the new representation and applying the according error propagation to the filters state covariance. Note that the filter is not reseted. Instead the propagated covariance is used for the following estimation stage.

#### Second Stage Calibration

In the second stage, the filters pose is chosen to be the IMU coordinate transformation $\Gamma_n^i$. The filter state is given by the local navigation filter state (see (4.2.8), page 77), except for exclusion of $t_{en}^e$ and the landmarks. For prediction the inertial model (4.2.21) is used, as is done for the SLAM filter in section 4.2.5. When updating the system state by the detected 2D/3D correspondences, the augmented ES observation model (see equation (A.2.8)), substituting frame $e$ by $n$, is used. Note that in contrast to the first stage, the IMU-camera calibration is now included in the update equation exclusively. By this, the IEKF optimally exploits the image measurements for the explicite estimation of $\Gamma_c^i$, in contrast to the implicit estimation in the first stage.

### Conclusion

The proposed calibration algorithm is capable of estimating even large misalignments (up to 40° for the orientation $\Phi_i^c$) in real-time. It provides precise estimates and the according covariance matrices. Moreover, the algorithm is performed fully automatic and provides evaluation information, that allows for online detection of erroneous estimations. Additionally, the estimation quality can be verified in parallel to the estimation procedure. This allows for adopting the used trajectory and it's length to reach an optimal calibration result. A more detailed description and evaluation of the proposed calibration software is given in [PK12b].

Chapter 7

# Experiments

In this chapter experiments are discussed, that have been made using different datasets. At first an evaluation on synthetic data is carried out, to determine the quality of the estimation processes. After the evaluation of the system models (initialization, map representation etc., see chapter 5) these tests aim at the applicability of the models and parametrizations to Kalman Filters.

Following that, two hardware platforms are used to apply the developed SLAM system to real world environments. The first experimental series was performed offline using a dataset, kindly provided by the "Rawseeds Project" (see [Cer+09], [Bon+06]). For the second series, the proposed software system has been applied online to a free-hand stereo-IMU setup. The experiments are evaluated in terms of estimation quality, consistency and real-time capability. Additionally the adjustment system proposed for PD SLAM is validated on the offline dataset.

All tests were performed on an Intel® Core™i7-3820 CPU @ 3.60GHz based desktop PC, using a NVIDIA GForce 285GTx® graphic card (used for SIFT feature detection, see section 2.5).

As remark it is stated, that all experiments have been carried out using an extended Kalman Filter EKF. A comparison to the IEKF is done in section 7.1.3. From the results discussed there, it is clear, that the EKF is the best choice for a general filter setup.

## 7.1 Synthetic Dataset

The estimations quality is a combination of the magnitude of estimation errors and their consistency with the estimated covariances. Since the relative importance between both characteristics depends on the application area,

their rating has to be done separately. To evaluate the quality with respect to the landmark parametrization, synthetical trajectories and according 3-space point clouds have been generated randomly. These poses and points were used to create ground truth pose increments and stereo correspondences for the predictions and updates of the used Kalman Filter. Following this, Gaussian noise was added to the synthetic data, by using the covariances $C_p$ for the pose increments and $C_{xy}$ for the stereo correspondences. By this, consistent predictions and measurements are available along with their ground truth values. Using this input, the landmarks parametrizations can be evaluated in consistency and quality without interference of wrong model assumptions (i. e., prediction models and noise assumptions).

As state model for this synthetical tests, a reduced filter state is used. Only the systems position and orientation are parametrized. The generated predictions are used as additive increments. The states for ES and ID are given by

$$p \quad = \quad \begin{pmatrix} t_{ec}^e & \Phi_c^e & \mathcal{X}_1 & \cdots & \mathcal{X}_{\mathtt{N}} \end{pmatrix} \tag{7.1.1}$$

The PD parameters include the pose $\Gamma_n^e$ of the navigation frame and the pose $\Gamma_c^n$ of the camera with respect to the navigation frame. That is:

$$p^{\mathrm{PD}} \quad = \quad \begin{pmatrix} t_{nc}^n & \Phi_c^n & t_{en}^e & \Phi_n^e & \mathcal{X}_1^{\mathrm{PD}} & \cdots & \mathcal{X}_{\mathtt{N}}^{\mathrm{PD}} \end{pmatrix} \tag{7.1.2}$$

The prediction uses the local pose increments $t_{nc}^n$ in the local navigation frame. This is similar to the prediction model, used for the inertial prediction model, introduced in section 4.2.5 and page 79. Thus, the analysis results can be easily transferred to the full system model.

The state parametrizations and prediction increments determine the prediction model

$$\begin{pmatrix} t_{en}^e \\ \Phi_n^e \\ \mathcal{X}_{1\ldots\mathtt{N}} \end{pmatrix} \quad \mapsto \quad \begin{pmatrix} t_{en}^e + \Phi_n^e \cdot t_{nc}^n \\ \Phi_n^e \cdot \Phi_c^n \\ \mathcal{X}_{1\ldots\mathtt{N}} \end{pmatrix} \tag{7.1.3}$$

for ES and ID representations. For the local PD representation, the predic-

tion is done by

$$\begin{pmatrix} 0 \\ 1^3 \\ t^e_{en} \\ \Phi^n_e \\ \mathcal{X}^{\mathrm{PD}}_{\mathtt{i}} \end{pmatrix} \mapsto \begin{pmatrix} t^n_{nc} \\ \Phi^n_c \\ t^e_{en} \\ \Phi^e_n \\ \mathcal{X}^{\mathrm{PD}}_{1\ldots\mathtt{N}} \end{pmatrix} \tag{7.1.4}$$

Note that due to the repetitive relocalization for PD (see section 4.2.7) the preceeding pose is always $\langle 1^3, 0_3 \rangle$. Moreover, the global pose $\Gamma^e_n$ is not effected by the prediction step. This is done, when the PD state is relocalized.

The trajectories for evaluation of the estimations quality were generated randomly using different characteristics. For example the average amount of pose increments for prediction, the process and observation noise, the number of used landmarks etc. are varied for different tests. For each setting of characteristics a series of estimation runs is performed, and the respective results are averaged for statistical stability.

As already stated, the evaluation of quality is done for the estimation error and their covariances with respect to the known ground truth. The most important measures of the errors magnitude are the root mean square errors (RMSE). Since the test results are averaged over a test series, also their standard deviations are of interest. Providing information on the estimation errors variability, they are a measure on the systems stability. For a fixed RMSE, decreased standard deviations imply decreased extremal magnitudes in the estimation error.

The estimations consistency is evaluated by analyzing the informativity measures for each estimated time step (see section 3.4.2). This is in accordance to the tests applied in the model evaluation chapter 5. To allow for applying these measures to a sequential estimation, a simplification has to be made. Estimating the random process $\hat{\wp} = \hat{p}_1, \hat{p}_2, \ldots$, results in a sample $\wp = p_1, p_2, \ldots$ of this process. For applying statistical test to this estimation series, the full covariance $C_\wp = \left(C_{p_\mathtt{i}, p_\mathtt{j}}\right)$ is needed. This is due to the fact, that the estimations $p_\mathtt{i}$ are correlated over time. Since the computation of $C_\wp$ and the respective error norms is hardly possible for long time series, only the diagonal elements $diag\,(C_\wp)$ are used. To reduce the impact of the

neglected correlations, also the consistency measures are averaged over time.

### 7.1.1 Pose Estimation

The first tests aim at the evaluation of precision, stability and consistency of pose estimation depending on the prediction quality. That is, two different process noise settings are used for pose prediction on the same trajectories. In the second test series, the influence of the number of used landmarks on the estimation process is evaluated.

The test series, discussed in this section, are generated using a trajectory created by a position and orientation random walk process. That is, local position and orientation increments are generated using normal distributed random variables. To simulate a meaningful movement, the position increments are generated with a mean of 2 baselines in $\mathsf{e}_3$ direction of the local camera frame $c$ (forward movement). Note that the forward movement is given in the local camera frame. Thus, the global movement is not necessarily in direction of the initial view.

The setup for the tests differ in the amount of process noise used for pose prediction. The first test series S1 uses standard deviations of $0.1[b]$ and $0.1°$ for the position and orientation prediction respectively. Series S2 generates the pose prediction for a process noise of $0.25[b]$ and $1°$ respectively. Both series average 500 test runs, having 1000 images (estimations) each.

#### Precision and Stability

In tables 7.1 and 7.2 the average RMSE $\bar{\epsilon}$ and their standard deviations for the estimated trajectories are given. The first table represents series S1, the second S2. To determine the average RMSE, for each of the 500 test runs $\wp_{\mathtt{i}}$, the RMSE $\epsilon_{\mathtt{i}}$ for the complete trajectory is computed. The given standard deviations, are computed over all $\epsilon_{\mathtt{i}}$ for the respective test series. They are given in percent of the average trajectory RMSE $\bar{\epsilon}$.

In both test scenarios the ID and PD representations clearly outperform the ES estimator. For all it's estimated components the average RMSE and it's standard deviation are significantly increased, compared to ID and PD. A noticeable fact is, that the ESs RMSE in $t^e_{ec(3)}$ is increased by 10%, compared to $t^e_{ec(1)}$ and $t^e_{ec(2)}$. This component corresponds to the camera

**Table 7.1.** Comparison of average root mean square errors and their standard deviations for trajectory estimation, setup S1. Averaged over 500 randomly generated trajectories with an overall length of $\sim$ 2000 baselines $b$ each. Series S1 using standard deviations $\sigma_t = 0.1[b]$ and $\sigma_\Phi = 0.1^\circ$ for position and orientation prediction respectively.

| RMSE for | | $t^e_{ec}(1)\,[b]$ | $t^e_{ec}(2)\,[b]$ | $t^e_{ec}(3)\,[b]$ |
|---|---|---|---|---|
| ES | mean | 20.4 | 19.0 | 22.0 |
| | stdev. | 87% | 86% | 71% |
| ID | mean | 12.7 | 12.3 | 12.7 |
| | stdev. | 71% | 68% | 67% |
| PD | mean | 10.4 | 9.45 | 10.0 |
| | stdev. | 66% | 66% | 69% |

| RMSE for | | $\Phi^e_c(1)\,[^\circ]$ | $\Phi^e_c(2)\,[^\circ]$ | $\Phi^e_c(3)\,[^\circ]$ |
|---|---|---|---|---|
| ES | mean | 2.39 | 2.39 | 2.18 |
| | stdev. | 64% | 62% | 56% |
| ID | mean | 1.48 | 1.52 | 1.40 |
| | stdev. | 42% | 40% | 39% |
| PD | mean | 1.24 | 1.24 | 1.18 |
| | stdev. | 51% | 52% | 49% |

rigs initial movement direction. Moreover, the inconsistencies in the ESs depth representation (see section 5.1.2, pages 101ff) are mainly in the rigs view direction. Thus, this invalid modeling especially effects $t^e_{ec(3)}$ for the first 50 to 100 estimations (afterwards the orientation changes due to the trajectories random walk).

The precision of estimation in ID is significantly improved for all estimated components, compared to ES. For both test series, ID is able to reduce the RMSE by $> 30\%$ in position and orientation. Moreover, the RMSEs percentaged standard deviations are smaller for ID than for ES. This leads to the conclusion, that the estimation is more stable, indicating a reduced risk of filter divergency. Actually, as discussed in section 5.1.2, the ES representation has proven to be highly unstable for high camera

**Table 7.2.** Comparison of average root mean square errors and their percentaged standard deviations for trajectory estimation, setup S2. Expect for the prediction noise, the test setup is the same as for S1. Prediction noise: $\sigma_t = 0.25[b]$ and $\sigma_\Phi = 1^\circ$.

| RMSE for | | $t^e_{ec(1)}\,[b]$ | $t^e_{ec(2)}\,[b]$ | $t^e_{ec(3)}\,[b]$ |
|---|---|---|---|---|
| ES | mean | 31.7 | 29.4 | 35.2 |
| | stdev. | 77% | 75% | 76% |
| ID | mean | 21.1 | 19.8 | 21.4 |
| | stdev. | 73% | 70% | 71% |
| PD | mean | 16.1 | 15.4 | 16.8 |
| | stdev. | 71% | 68% | 69% |

| RMSE for | | $\Phi^e_{c(1)}\,[^\circ]$ | $\Phi^e_{c(2)}\,[^\circ]$ | $\Phi^e_{c(3)}\,[^\circ]$ |
|---|---|---|---|---|
| ES | mean | 3.80 | 3.78 | 3.38 |
| | stdev. | 50% | 54% | 49% |
| ID | mean | 2.55 | 2.60 | 2.38 |
| | stdev. | 40% | 41% | 39% |
| PD | mean | 2.04 | 2.06 | 1.89 |
| | stdev. | 39% | 41% | 39% |

landmark distances. For the tests carried out, it was observed, that landmarks with initial disparities of less than 3 times the observations standard deviation, frequently cause divergence of the ES filter. In contrast to that, the depth and disparity estimation in ID and PD respectively, have proven to be robust to such far landmarks. This is due to the correct initialization of the distances uncertainties by stereo correspondences.

The PD representation clearly performs best with respect to the RMSE among the discussed models. Compared to ES, the error is reduced by 50% and 45% for the position and orientation respectively for both test series. Compared to ID, the reduction is about 20% of the averaged RMSE. Moreover, the RMSEs percentaged standard deviation is reduced significantly compared to ES. Except for the orientation in test series S1, it is approximately the same as for ID. Since the standard deviations absolute values are comparable for ID and PD in this special case, the stability for

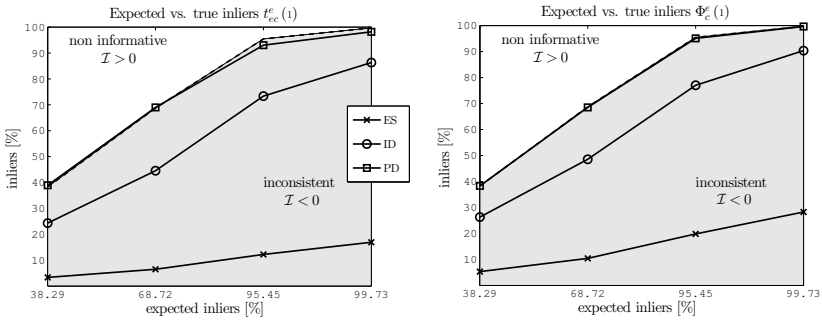


**Figure 7.1.** Visualization of informativity for first position $t^e_{ec}(1)$ and orientation $\Phi^e_c(1)$ components (all representations show their best results for these). Test setup S1, compare table 7.3.

both models can be considered, to be quite similar.

### Consistency

To evaluate the consistency for the test setups S1 and S2, the average informativity for all estimated pose components has been determined. As stated above, the estimated state sequence cannot be tested on the whole. Thus, the state components are treated individually. Moreover, their informativity is determined for each estimation step in all test series. That is, an average value for 500×1000 estimations is computed. The averaged informativity measures for both test series are given in tables 7.3 and 7.4 respectively. Additionally, figure 7.1 visualizes the informativity measures for the first position and orientation parameters of setup S1.

As for the trajectories RMSE, the ID and PD representations clearly outperform the ES estimator in terms of informativity. Keeping in mind the bounds for $\mathcal{I}$ (see equation (3.4.8), page 55) it can be seen, that the informativity for ES is close to it's lowest possible values. This becomes more clear in figure 7.1. Neither the estimated positions nor the orientations standard deviations give a valuation of the true trajectory error. As discussed in section 5.1 (pages 100ff), the initialization of ES landmarks results in a highly inconsistent state covariance. Thus, during estimation the linear

**Table 7.3.** Informativity for pose estimations on synthetical data, averaged of 500 randomly generated trajetories. $\sigma_t = 0.1[b]$, $\sigma_\Phi = 0.1^\circ$

| | | $t^e_{ec}(1)$ | $t^e_{ec}(2)$ | $t^e_{ec}(3)$ | $\Phi^e_c(1)$ | $\Phi^e_c(2)$ | $\Phi^e_c(3)$ |
|---|---|---|---|---|---|---|---|
| ES | $\mathcal{I}_{0.5}$ | -35 | -35 | -37 | -33 | -33 | -31 |
| | $\mathcal{I}_1$ | -62 | -62 | -66 | -58 | -59 | -54 |
| | $\mathcal{I}_2$ | -83 | -83 | -90 | -76 | -76 | -68 |
| | $\mathcal{I}_3$ | -83 | -82 | -91 | -71 | -72 | -61 |
| ID | $\mathcal{I}_{0.5}$ | -14 | -15 | -23 | -12 | -13 | -11 |
| | $\mathcal{I}_1$ | -24 | -25 | -37 | -20 | -22 | -19 |
| | $\mathcal{I}_2$ | -22 | -23 | -34 | -18 | -20 | -16 |
| | $\mathcal{I}_3$ | -13 | -14 | -17 | -9 | -11 | -8 |
| PD | $\mathcal{I}_{0.5}$ | 1 | 6 | -10 | 0 | -1 | 1 |
| | $\mathcal{I}_1$ | 0 | 3 | -17 | 0 | -2 | 0 |
| | $\mathcal{I}_2$ | -2 | -1 | -11 | 0 | 0 | 0 |
| | $\mathcal{I}_3$ | -2 | -1 | -3 | 0 | 0 | 0 |

**Table 7.4.** Informativity for pose estimations on synthetical data, averaged of 500 randomly generated trajetories. $\sigma_t = 0.25[b]$, $\sigma_\Phi = 1^\circ$

| | | $t^e_{ec}(1)$ | $t^e_{ec}(2)$ | $t^e_{ec}(3)$ | $\Phi^e_c(1)$ | $\Phi^e_c(2)$ | $\Phi^e_c(3)$ |
|---|---|---|---|---|---|---|---|
| ES | $\mathcal{I}_{0.5}$ | -36 | -36 | -37 | -34 | -34 | -32 |
| | $\mathcal{I}_1$ | -64 | -64 | -67 | -60 | -60 | -57 |
| | $\mathcal{I}_2$ | -86 | -86 | -91 | -79 | -79 | -73 |
| | $\mathcal{I}_3$ | -86 | -86 | -93 | -76 | -76 | -68 |
| ID | $\mathcal{I}_{0.5}$ | -24 | -23 | -28 | -21 | -21 | -21 |
| | $\mathcal{I}_1$ | -41 | -39 | -48 | -35 | -36 | -35 |
| | $\mathcal{I}_2$ | -45 | -43 | -54 | -37 | -38 | -37 |
| | $\mathcal{I}_3$ | -34 | -31 | -41 | -25 | -26 | -26 |
| PD | $\mathcal{I}_{0.5}$ | 1 | 3 | -13 | -3 | -4 | -2 |
| | $\mathcal{I}_1$ | -2 | -1 | -19 | -5 | -6 | -4 |
| | $\mathcal{I}_2$ | -4 | -3 | -14 | -3 | -3 | -2 |
| | $\mathcal{I}_3$ | -2 | -1 | -4 | 0 | -1 | -0 |

estimator assumes a too high weighting for the filter state with respect to the observations (see also 7.1.3 for a more detailed description). This results in a reduction of pose estimation quality and, in succession, more inconsistent covariances.

Compared to ES, the ID estimator improves the informativity of the predicted covariances significantly. The improved consistency of the landmark representation is capable, of halving the informativity measure. The orientation's $3\sigma$-bound, which is mostly used for precision forecast, for test setup S1 gives a notion on the expectable magnitude of estimation error. Nevertheless, the predicted $\sigma$-bounds are not precise enough, to allow for a reliable error appraisement for all components.

In contrast to ES and ID, the PD representation provides informativity of high quality, except for $t^e_{ec}(3)$. The absolute values of the informativity measures for the $3\sigma$-bounds are negligible, i. e., $|\mathcal{I}_3| \leqslant 2$. Even though small inconsistencies for the lower $\sigma$-bounds occur, the estimated covariance provides a meaningful valuation for the estimation error. With an informativity measure of $-3$ and $-4$ for the $3\sigma$-bounds, the maximal error bounds are still met for $t^e_{ec}(3)$. Nevertheless, the lower informativity measures show certain inconsistencies. The improvement in consistency for PD compared to ES and ID is especially visible in figure 7.1. As can be seen, the PD representation meets the true expected percentage of inliers. The ES representation is obviously inconsistent. Although the consistency is significantly improved with respect to ES, the ID parametrization provides only a rough guess for the true error bound.

It is worth mentioning, that the informativity for all parametrizations degrades, for increasing orientation uncertainties $\geqslant 1°$ of the prediction noise. This is mainly caused by the violation of assumptions, made for rotation linearization, see (2.4.5) and (2.4.6) on page 32. Since the linearization error strongly increases for angles $> 1°$, the error propagation becomes increasingly unstable. Thus, this phenomenon is mostly resided in the pose representation and to a lesser extent in the landmark models.

In tables 7.3 and 7.4 it can be seen, that all estimators have the poorest informativity in the position $t^c_{ec}(3)$. This leads to the conclusion, that the initials camera view direction $\mathsf{e}_3$, which is identical for all tests, introduces a correlation between the test runs. Whether these are correlations caused by the generated data or due to the estimation process, has to be analyzed

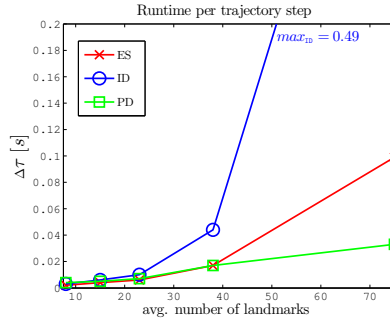


**Figure 7.2.** Runtime $\Delta\tau$ for a trajectory step, depending on the average number of used landmarks. Runtime includes prediction, insertion of landmarks and Kalman update for ES, ID and PD. For PD, the relocalization is also included.

in future work.

### Quality vs Complexity

For relating the quality of estimation to the computational effort, additional test series have been carried out. The generated trajectories are similar to those for S1 in the previous section. The impact of different numbers of active landmarks to the estimation process is evaluated by comparing the estimations RMSE and runtime for all representations.

The average runtime for a single filter step is visualized in figure 7.2. For ES and ID the runtime includes the filter prediction, Kalman update and initialization of new landmarks. Beside these, the PDs runtime additionally comprises the relocalization process. The figure includes a test using an average of 75 landmarks. Since this estimation sequence has a high expenditure of time, no averaging over a long test series has been carried out. Thus, no statistical meaningful information for an analysis of the trajectories RMSEs is available.

As can be seen, the runtime for all parametrizations is similar for up to $\leqslant 23$ landmarks. Increasing the number of landmarks to 38, ES and PD have a similar gain in runtime of $\sim 2\times$. In contrast to that, it increases by a factor of 5 for the ID parametrization. Increasing the number of landmarks

further, results in a runtime scale of 10 for ES and ID parametrizations. On the other hand, the PD parametrization doubles it's computational effort in time.

For ES, the gain is caused by the increasing state size. Since newly introduced landmarks require a full covariance propagation (see section 4.2.4) on the augmented state, the growing state impacts the time, needed for initialization and update. The ID landmarks use 6 parameters instead of 3. Moreover, the complexity of the covariance propagation grows quadratic in the state size. Thus, the impact to ID is much higher compared to ES and PD. Thanks to the simple initialization procedure of PD, the time for insertion of landmarks is negligible. Nevertheless, the relocalization procedure has to be applied, before the next update can be performed. Since this is done after the lost landmarks have been deleted, the respective state is not as large as the augmented, which contains the new landmarks. Thus, the relocalization in PD takes significantly less time, compared to the state augmentation for ES and ID.

It has to be stated, that these relations depend, on how often new landmarks have to be inserted. For the used setup, each landmark has an average of 6 applied updates. Using trajectories with other landmark tracking characteristics, will influence the ratios. But, in essence, the high computational effort for ID with large state sizes will remain, as stated before by Montiel et al. in [CDM08].

In table 7.5 and figure 7.3 the trajectories RMSE in dependence on the average number of landmarks, used for each Kalman update, is given. As can be seen, the RMSE for ID and PD scale similarly with the number of used landmarks. In especially, the estimation error decreases with an increasing number of landmarks.

In contrast to that, the ES parametrization performs best for an average of 15 active landmarks. On the one hand, fewer landmarks cause a loss of available information on the observed scene. On the other hand, the inconsistency of ES landmarks increases, when applying Kalman updates. This can be concluded from a comparison of the results for the informativity measure of landmark initialization (table 5.1 and 5.3, pages 103, 107) and landmark estimation (see table 7.7 in the following section). Moreover, the inconsistent observation prediction (see section 5.3.1, page 118) causes the estimator to assess them as too high with respect to the observation. Since

**Table 7.5.** Comparison of root mean square error for trajectory estimation depending on the average number #pts of used landmarks. The number of landmarks is regulated by limiting the active landmarks to at most 10, 20, 30 and 50 respectively. Test setup as in series S1 (see tables 7.3, 7.1).

| | #pts | $t^e_{ec}(1)$ [$b$] | $t^e_{ec}(2)$ | $t^e_{ec}(3)$ | $\Phi^e_c(1)$ [°] | $\Phi^e_c(2)$ | $\Phi^e_c(3)$ |
|---|---|---|---|---|---|---|---|
| ES | 8 | 19 | 19 | 20 | 2.4 | 2.3 | 2.3 |
| | 15 | 17 | 17 | 22 | 2.0 | 2.1 | 1.9 |
| | 23 | 18 | 19 | 23 | 2.2 | 2.1 | 2.0 |
| | 38 | 19 | 20 | 24 | 2.3 | 2.4 | 2.1 |
| ID | 8 | 14 | 16 | 16 | 1.4 | 1.6 | 1.6 |
| | 15 | 13 | 12 | 13 | 1.6 | 1.5 | 1.4 |
| | 23 | 11 | 12 | 10 | 1.2 | 1.2 | 1.2 |
| | 38 | 8.3 | 8.8 | 9.4 | 0.9 | 1.0 | 1.0 |
| PD | 8 | 12 | 12 | 12 | 1.6 | 1.5 | 1.5 |
| | 15 | 9.9 | 9.5 | 10 | 1.2 | 1.2 | 1.2 |
| | 23 | 8.5 | 8.9 | 8.6 | 1.0 | 1.0 | 0.9 |
| | 38 | 7.4 | 7.6 | 9.1 | 0.9 | 0.9 | 0.8 |

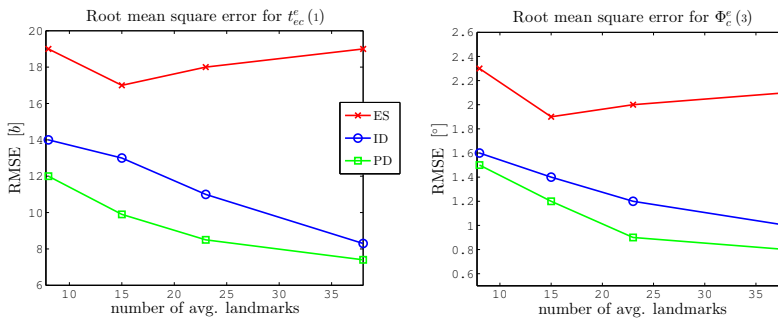


**Figure 7.3.** Root mean square error of first position and third orientation component versus the average number of used landmarks. According to table 7.5.

this effect is amplified when increasing the number of landmarks, the gain of information is eliminated by the erroneous weighting.

Considering the estimator's runtime and the quality of estimation, a number of 20 to 25 landmarks has proven to be an appropriate setting. On the one hand, the gain in runtime for the ID parametrization for $\geqslant 30$ landmarks is not justifiable. On the other hand, the trajectories RMSE for PD is not reduced significantly, using more than 25 landmarks.

### 7.1.2 Landmark Estimation

In the following, the estimation's quality and consistency for landmarks is analyzed. Therefore, the estimators are applied to the circle segment trajectory, used in section 5.3.1 (page 118) for evaluation of the observation models. As test landmarks 20 3-space points were distributed in a sphere around the circles center. The tests were carried out with three different setups, corresponding to the analysis of the observation prediction.

In tables 7.6 and 7.7 the landmarks estimations RMSE and informativity measures respectively are given. To be able to compare the RMSEs, they are computed for the reconstructed 3-space points. The informativity is given in parameter space, since no valid error propagation to 3-space is possible.

The results for the RMSEs fit the analysis of observation prediction, made in section 5.3 (page 116). For a traveled circle segment of 90° with a single update, the ID and PD estimators end up with a high error, compared to ES. This is due to the observation prediction's high linearization error for ID and PD, discussed in the model evaluation section. Decreasing the change in view direction (see multiple update and 45° tests in table 7.6), ID and PD clearly outperform the ES representation. Due to the inconsistencies of landmark representation and high linearization errors for low view direction changes, the ES estimator degrades strongly. Moreover, the improved linearization in ID and PD improves the estimation quality significantly.

From the informativity analysis (see table 7.7) it follows, that, compared to the RMSE, a consistent covariance estimation is more sensible to linearization errors. Not until the change in view direction is $\leqslant 22.5°$ between two updates, the ID and PD estimators are consistent. Nevertheless, the ID representation suffers from a significant loss of informativity. Comparing

**Table 7.6.** Root mean square error of the 3-space points computed from the landmarks estimated using a circle segment trajectory (same as figure 5.7, page 118) around the 20 3-space points. Point camera distance is 15 to 25 baselines. Top table uses a 90° circle segment, bottom table uses 45°. Values given for different prediction and update step combinations.

90° circle segment

| RMSE for | 1 pred. 1 upd. | | | 8 pred. 2 upd. | | | 8 pred. 8 upd. | | |
|---|---|---|---|---|---|---|---|---|---|
| | $X_{(1)}$ | $X_{(2)}$ | $X_{(3)}$ | $X_{(1)}$ | $X_{(2)}$ | $X_{(3)}$ | $X_{(1)}$ | $X_{(2)}$ | $X_{(3)}$ |
| ES | 0.1 | 0.1 | 0.5 | 6.3 | 0.4 | 3.8 | 5.5 | 1.1 | 8.7 |
| ID | 0.2 | 0.2 | 2.5 | 0.1 | 0.1 | 0.6 | 0.1 | 0.1 | 0.3 |
| PD | 0.2 | 0.2 | 2.5 | 0.1 | 0.4 | 0.5 | 0.1 | 0.2 | 0.4 |

45° circle segment

| RMSE for | 1 pred. 1 upd. | | | 8 pred. 2 upd. | | | 8 pred. 8 upd. | | |
|---|---|---|---|---|---|---|---|---|---|
| | $X_{(1)}$ | $X_{(2)}$ | $X_{(3)}$ | $X_{(1)}$ | $X_{(2)}$ | $X_{(3)}$ | $X_{(1)}$ | $X_{(2)}$ | $X_{(3)}$ |
| ES | 0.4 | 0.3 | 5.8 | 2.7 | 0.3 | 4.7 | 3.3 | 1.0 | 8.3 |
| ID | 0.1 | 0.1 | 0.6 | 0.1 | 0.1 | 0.6 | 0.1 | 0.1 | 0.4 |
| PD | 0.1 | 0.1 | 0.6 | 0.3 | 0.4 | 0.6 | 0.2 | 0.1 | 0.4 |

**Table 7.7.** Informativity of ES, ID and PD landmarks according to table 7.6. Informativity is given in the respective landmarks parameter space.

90° circle segment

| | 1 pred. 1 upd. | | | 8 pred. 2 upd. | | | 8 pred. 8 upd. | | |
|---|---|---|---|---|---|---|---|---|---|
| | $\mathcal{I}_1$ | $\mathcal{I}_2$ | $\mathcal{I}_3$ | $\mathcal{I}_1$ | $\mathcal{I}_2$ | $\mathcal{I}_3$ | $\mathcal{I}_1$ | $\mathcal{I}_2$ | $\mathcal{I}_3$ |
| ES | -56 | -65 | -53 | -69 | -95 | -99 | -67 | -91 | -91 |
| ID | -69 | -95 | -99 | -50 | -55 | -41 | 16 | 3 | 0 |
| PD | -69 | -95 | -99 | -32 | -25 | -9 | 1 | -1 | 0 |

45° circle segment

| | 1 pred. 1 upd. | | | 8 pred. 2 upd. | | | 8 pred. 8 upd. | | |
|---|---|---|---|---|---|---|---|---|---|
| | $\mathcal{I}_1$ | $\mathcal{I}_2$ | $\mathcal{I}_3$ | $\mathcal{I}_1$ | $\mathcal{I}_2$ | $\mathcal{I}_3$ | $\mathcal{I}_1$ | $\mathcal{I}_2$ | $\mathcal{I}_3$ |
| ES | -69 | -95 | -99 | -68 | -94 | -97 | -61 | -77 | -69 |
| ID | -68 | -92 | -93 | 13 | 3 | 0 | 13 | 3 | 0 |
| PD | -69 | -95 | -98 | -2 | 0 | 0 | -2 | -1 | 0 |

**Table 7.8.** Root mean square error of IEKF (5 iterations) for ES, ID and PD parametrizations. Additionally the IEKF error is given as percentage of RMSE for EKF. The test setup is S1 with low prediction noise ($\sigma_t = 0.1[b]$, $\sigma_\Phi = 0.1^\circ$), for comparison see table 7.1 (page 157).

| RMSE for | | $t^e_{ec}(1)\,[b]$ | $t^e_{ec}(2)\,[b]$ | $t^e_{ec}(3)\,[b]$ |
|---|---|---|---|---|
| ES | mean | 21.0 | 20.4 | 29.1 |
| | % of EKF | 103% | 107% | 132% |
| ID | mean | 22.7 | 24.7 | 21.6 |
| | % of EKF | 179% | 332% | 179% |
| PD | mean | 10.2 | 9.4 | 9.9 |
| | % of EKF | 99% | 100% | 99% |

| RMSE for | | $\Phi^e_c(1)\,[^\circ]$ | $\Phi^e_c(2)\,[^\circ]$ | $\Phi^e_c(3)\,[^\circ]$ |
|---|---|---|---|---|
| ES | mean | 2.47 | 2.47 | 2.16 |
| | % of EKF | 103% | 103% | 99% |
| ID | mean | 3.08 | 2.98 | 2.64 |
| | % of EKF | 208% | 196% | 188% |
| PD | mean | 1.25 | 1.25 | 1.18 |
| | % of EKF | 101% | 100% | 100% |

this to the results in section 5.3 implies, that even a small linearization error in the observation prediction, can cause a strongly degraded consistency.

### 7.1.3 EKF versus IEKF

Since the models, used for Kalman Filter estimation, are non linear, the usage of iterated Kalman Filters (see section 3.3.3) suggests itself. To analyze the effect of this filter specialization on the discussed problem, it has been applied to the test setups S1 and S2 (see section 7.1.1, page 156).

In tables 7.8 and 7.9 the average error of estimation for test setup S1 and S2 respectively are given. In addition to the absolute trajectories RMSEs (computed as in 7.1.1), their fraction in percent of the respective RMSE of the EKF estimation is given. This eases the comparison between

**Table 7.9.** Root mean square error of IEKF (5 iterations) for ES, ID and PD parametrizations. Additionally the IEKF error is given as percentage of RMSE for EKF. The test setup is S2 with high prediction noise ($\sigma_t = 0.25[b]$, $\sigma_\Phi = 1^\circ$), for comparison see table 7.2 (page 158).

| RMSE for | | $t^e_{ec}(1)\,[b]$ | $t^e_{ec}(2)\,[b]$ | $t^e_{ec}(3)\,[b]$ |
|---|---|---|---|---|
| ES | mean | 32.5 | 33.1 | 47.3 |
| | % of EKF | 103% | 113% | 134% |
| ID | mean | 26.6 | 34.4 | 24.9 |
| | % of EKF | 126% | 173% | 117% |
| PD | mean | 15.4 | 14.9 | 16.5 |
| | % of EKF | 95% | 97% | 98% |

| RMSE for | | $\Phi^e_c(1)\,[^\circ]$ | $\Phi^e_c(2)\,[^\circ]$ | $\Phi^e_c(3)\,[^\circ]$ |
|---|---|---|---|---|
| ES | mean | 4.21 | 4.15 | 3.61 |
| | % of EKF | 111% | 110% | 107% |
| ID | mean | 3.53 | 3.51 | 3.14 |
| | % of EKF | 138% | 135% | 132% |
| PD | mean | 1.97 | 1.96 | 1.82 |
| | % of EKF | 95% | 97% | 98% |

both test applications.

As can be seen, the error for S1 and S2 for the PD filter does not change or is slightly reduced respectively. In contrast to that, for ES the RMSE is increased by $\frac{1}{3}$ for some components. Comparing the IDs RMS errors for EKF and IEKF filters, the inapplicability of an iterated Kalman Filter to ID estimation is obvious. The errors in all components increase significantly. For $t^e_{ec}(2)$ in S1 it is tripled. For clarifying this phenomenon, recall the consistency evaluation for pose (see section 7.1.1, page 159) and landmark (see previous section) estimations. For ES, all estimation where inconsistent with a similar informativity measure $\mathcal{I} << 0$. For a valid linearization area, the ID poses covariance was determined as too small, i. e., $\mathcal{I} < 0$, whereas the landmarks are consistent but non informative, i. e., $\mathcal{I} > 0$. In contrast to that, all estimated components for PD are consistent and informative,

i. e., $\mathcal{I} \sim 0$.

In ES, the inconsistency of $C_{\text{ES}} < \hat{C}_{\text{ES}}$ causes a wrong weighting during the Kalman Update (see (3.3.32), page 50). That is, the observation predictions covariance is too 'small', i. e., $HC_{\text{ES}}H^T < H\hat{C_{\text{ES}}}H^T$ (see observation analysis section 5.3, page 118). By this, the Kalman gain matrix uses a decreased weight for the contradiction $z = y - h^{\text{ES}}(p)$. Due to this, the impact of the observed stereo correspondences is reduced and the estimation tends to be too close the a priori state. Consequently, the penalty term $H^{(\imath)} \cdot \left(p_0 - p^{(\imath)}\right)$, included in $z^\imath$ for the iterated filter, is also reduced in weight. By this, iterating the estimation amplifies the wrong weighting and degrades the iterated result.

In contrast to ES, a non informative observation prediction (see 5.3) is incorporated by ID. That is, the observed stereo correspondences are weighted too high, resulting in an increased weight for the contradiction $z = y - h^{\text{ES}}(p)$. Moreover, $C_{\text{ID}}$ introduces a wrong weighting between the a priori pose and the respective landmarks. This is due to the inconsistent pose estimate (high weight) and the non informative landmark estimate (low weight). Due to this wrong, state internal weighting the Kalman Gain wrongly distributes the contradiction. As for ES, this also effects the penalty term $H^{(\imath)} \cdot \left(p_0 - p^{(\imath)}\right)$, and by this, degrades the iterated results. Since the weighting between the pose and the landmarks is strongly influenced by the process noise, it has a major impact to the magnitude of degradation. By adding high process noise to the pose the filter components relative weighting gets less important. By this, increasing the process noise (compare table 7.8 and 7.9) reduces the degradation significantly for ID. A similar effect can be achieved by using an additional observation noise ($V_{\mathtt{k}}$ in section 3.3.3, page 50). It models the linearization error by a zero mean Gaußian noise. Since the involved covariance matrix has to be adopted to the amount of linearization error, it cannot be assumed as constant in general. Thus, tuning a filter for this model is hardly possible for the proposed estimators.

In contrast to ES and ID, the pose and landmark covariances are valid for PD. Moreover, the linear landmark initialization and the close to linear observation model allow for a precise linearization in the EKF. Thus, applying the IEKF neither degrades the estimation, nor improves it significantly.

## 7.2 Offline Dataset

To evaluate the applicability of the proposed SLAM systems to real world problems, this section analyzes their performance on a realistic, autonomous robot platform. For this task the "RawSeeds Database" dataset "Bicocca 2009-02-25b" is used. The datasets of this research project have been made public (see [Raw], [Cer+09], [Bon+06]), to enable the research community, to test algorithms on hardware platforms, which are not available in common. It is especially valuable since ground truth information of high precision is available, such that the estimation of real world experiments can be evaluated precisely. Due to this, it was used as reference dataset in many publications (e.g. see [Bur+09], [Civ+09], [Wil+09]).

As already stated, the RawSeeds Project dataset delivers ground truth information on the trajectory. Along with this data, error evaluation procedures are given, that use two error measures. These are the **A**bsolute **T**rajectory **E**rror ATE and the **R**elative **P**ose **E**rror for the position T-RPE and the orientation R-RPE. The former is given by the mean of absolute position errors (norm of displacement vector) for each ground truth pose. Moreover, the standard deviation of the absolute position errors is given. The T-RPE and R-RPE represent the mean absolute error of relative poses. I. e., the transformation between succeeding poses and it's deviation from ground truth is determined. To be comparable to system evaluations, performed by other researchers on the RawSeeds data, these error measures are provided in addition to the RMSE.

### 7.2.1 Hardware and Dataset

The provided dataset includes information from various types of sensors, as odometry, different cameras and more. Since this thesis aims at the parametrization of IMU-aided stereo SLAM, only the respective subset is used. The recorded data is made up of 26335 grayscale stereo image pairs (two Videre Design STH-DCSG-VAR, ©by Videre Design), captured with a frame-rate of 15Hz and a resolution of $640 \times 480$ pixels. In figure 7.4 two sample images, taken at the systems initial position, are depicted. In addition a low cost IMU (Xsens MTi ©by Xsens) provides measurements of accelerations and rates of turn at 128Hz, which are time aligned with the

**Figure 7.4.** Sample images from Rawseeds Database set "Bicocca" (2009-02-25b). Left/master- and right/slave-camera.

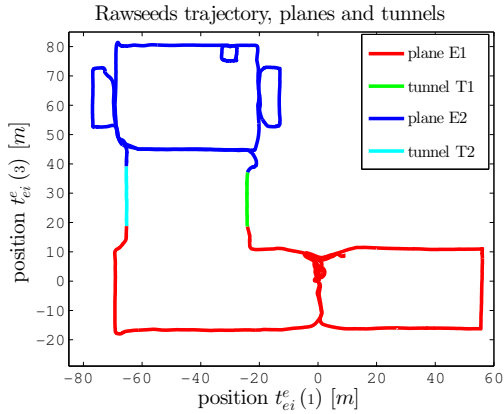


**Figure 7.5.** Visualization of the RawSeeds trajectory. E1 and E2 lie in a plain each. T1 and T2 connect the plains and are slightly inclined.

images by absolute timestamps.

The traveled trajectory, depicted in figure 7.5, is made up of various closed loops and is distributed over two floors (E1 and E2 in the figure). For both floors the trajectory lies in a plane, with both planes being connected by two small tunnels (T1 and T2) with a slight inclination. The overall

**Figure 7.6.** Sample images of the floor connectors T1 (left) and T2 (right) from the RawSeed dataset. Note the saturation and reflections on the windows walls and the floor.

length is round about 780 meters and 30 minutes.

The floor connections T1 and T2 are a special challenge for image processing systems. As can be seen in figure 7.6 the task of landmark matching is challenging due to the mirroring walls. As will be shown, the proposed system is able to detect missmatched mirror features, thanks to it's reliable prediction model. Moreover, for trajectory parts with a length of up to 250 images no visual features were present. Nevertheless, the system is able to cover such 'holes' nearly without loss of precision.

An internal and external calibration for the stereo cameras is provided along with the dataset. It is used, to undistort the detected visual features as discussed in section 2.4.1, page 26. In contrast to the tests in section 7.3 no data for application of the proposed camera to IMU calibration (see section 6.9, page 148) is available. Thus, the camera-IMU calibration $\Gamma_i^c$ from the dataset has to be used. Early tests suffered from biased poses for long term estimation. Due to this, the estimation of $\Gamma_i^c$ was included in the system state (see section 4.2.3 and appendix A.2), to compensate for small calibration errors.

### 7.2.2 Pose Estimation

In the following, the estimations quality is evaluated with respect to the filter tuning. Therefore, test runs were carried out, using different noise

and initializations setups. Due to the known sensibility of Kalman Filters to erroneous noise tuning, this has been the major testing field for pose estimation evaluation. The main issue is the process noise, since it has to cover improper modeling of the systems prediction. On the one hand, naive tuning has been applied for the prediction noise, obviously bearing the risk of tuning to the special dataset. On the other hand, the process noise has been deduced using the error approximations discussed in section 4.3.1, page 92.

The observation noise can be deduced from the quality of the used feature matching (see 2.5) and is set to 2 to 3 pixels for a resolution of $640 \times 480$. The noise parameters of the IMUs hardware (see section 4.3.2) is taken from the manufacturer information for the used IMU. The biases process noise, used to compensate for drift, are set to $10^{-4}[\frac{m}{s^2} \cdot \frac{1}{\sqrt{s}}]$ and $0.006[\frac{^\circ}{s} \cdot \frac{1}{\sqrt{s}}]$ (see section 3.1.4, page 44). These values have proven, to prevent the filter from getting stuck in estimated biases without losing informativity.

**Initialization** The initial state covariance (see section 6.2) can be determined from hardware specifications and environment conditions. The IMUs biases are assumed, to be not larger than $0.1[\frac{m}{s^2}]$ for the acceleration $b_a$ and $0.03[\frac{rad}{s}]$ (hardware specifications), thus $\sigma_a = 0.05[\frac{m}{s^2}]$ and $\sigma_w = 0.01[\frac{rad}{s}]$ are chosen. Since the robots initial position is used as absolute reference, it's standard deviation is chosen as 0 (perfectly known). As for the position, the heading $\Phi^i_{e(2)}$ (rotation for $\mathsf{e}_2$-axis , see section 2.2.2, page 14) is chosen as initial orientation, thus a standard deviation of $0^\circ$ is assumed.

For the velocity a small initial deviation is given ($0.01\frac{m}{s}$) to compensate for slight movements during startup. Since the initial robot orientation is known to be horizontally slightly misaligned, an uncertainty of a few degrees (standard deviation $\sim 4^\circ$) for $\Phi^e_{i(1)}$ and $\Phi^e_{i(3)}$ is assumed. The camera to IMU rotation $\Phi^c_i$ is set to identity and is subject to consistency analysis in section 7.2.3. The respective translation $t^c_{ci}$ is set to the given calibration and only a small error of a few centimeters is assumed.

The state initialization is straight forward. Except for the camera to IMU transformation $\langle \Phi^c_i, t^c_{ci} \rangle$ and the IMU scales $s_a$, $s_w$ all components are set to 0. The scales are unknown, thus set to neutral value of 1. As for the covariance initialization, $\Phi^c_i$ is tested using varying settings. The translation $t^c_{ci}$ is set, as is given in the dataset calibration.

**Table 7.10.** Plane navigation error analysis for S1 (heuristic process noise). Root mean square errors, absolute trajectory ATE and relative pose error RPE for the estimated trajectory over all 26335 poses.

| | RMSE | | |
|---|---|---|---|
| | $t^e_{ei(1)}$ $[m]$ | $t^e_{ei(3)}$ $[m]$ | $\Phi^e_{i(2)}$ $[^\circ]$ |
| ES | 1.445 | 1.669 | 2.580 |
| ID | 2.881 | 3.239 | 3.758 |
| PD | 2.087 | 1.559 | 1.874 |

| | RawSeeds error | | |
|---|---|---|---|
| | ATE $\bar{\varepsilon} \pm \sigma_\varepsilon$ | T-RPE $[m]$ | R-RPE $[^\circ]$ |
| ES | 2.0150 $\pm$ 0.9555 | 0.6839 | 0.8929 |
| ID | 3.4829 $\pm$ 2.5807 | 0.9196 | 1.0140 |
| PD | 1.9488 $\pm$ 1.7283 | 0.6727 | 0.8479 |

#### Heuristic Tuning

is often applied to Kalman Filtering, when it is to be used in similar environments and no qualitative information on the noise parameters is available. The difficulty is to find a balance between consistency, informativity and precision. In the following, an exemplarily setting will be discussed. After performing various test runs on this dataset and the free-hand system (see section 7.3), the noise parameters used here have proven to be a good setting. Note, that the parameters were tuned for the RS dataset and the free-hand system, to avoid dataset tuning. It assumes a random walk drift due to model errors of $1 \cdot 10^{-3}[\frac{m}{\sqrt{s}}]$ and $0.05[\frac{^\circ}{\sqrt{s}}]$ for the position and orientation respectively. This parameter setup is called S1 in the following.

**Plane Navigation** In table 7.10 the RMSE for each plane navigation component as well as the ATE and T-/R-RPEs are given. As can be seen, the ES representation performs best with respect to the position in the navigation plane. The PD estimation provides the lowest RMSE for the heading. Taking position and orientation for all error measures into account, the ES and the PD representations perform comparably. In contrast to this,

**Table 7.11.** Error analysis in final height $t^e_{ei(2)}$ for S1 (heuristic process noise). $\epsilon$ final error, $3\sigma$ confidence interval and RMSE for floor E1.

| | $\epsilon$ | $3\sigma$ | RMSE |
|---|---|---|---|
| ES | -9.0 | $\pm$ 0.733 | 6.746 |
| ID | -2.025 | $\pm$ 1.299 | 1.344 |
| PD | -1.754 | $\pm$ 1.315 | 1.218 |

the ID representation cannot achieve a similar estimation quality.

Further tests showed similar results for different parameter settings. Although the navigation in the plane is estimated with high precision by ES, it was found to be unstable in other components. This can be seen in the height estimation, and especially when the initial IMU-camera calibration is altered (see section 7.2.3).

**Height Estimation** Since no ground truth information on the systems height $t^e_{ei(2)}$ is available, it has to be analyzed individually. It is known, that the floor E1 (see figure 7.5) has a constant height. This information can be used for error analysis. On the one hand, the deviation $\epsilon$ from 0 for the estimated height and the respective $3\sigma$-bound can be compared. On the other hand, the heights RMSE for the sub-track in E1 can be determined.

These error measures are given in table 7.11. As can be seen, ID and PD end up with a similar error in height at the end of the trajectory. Although both are not consistent, their $3\sigma$-bounds provide an appraisement of the error to be expected. In contrast to this, the ES parametrization clearly ends up with an erroneous and inconsistent estimation. On the one hand, the final height error and the RMSE on E1 are increased by $> 300\%$ in comparison to ID and PD. On the other hand, the provided $3\sigma$-bound is halved. This loss of estimation quality annihilates the comparably high quality for plane navigation.

It has to be stated, that for all applied tests the height component has proven to be the most unstable parameter for all parametrizations. Moreover, the estimation error undergoes a constant drift. This is in opposition to the remaining estimations, that appear to undergo a random walk process (see figures 7.7, 7.8, 7.9). Nevertheless, the ES representation always ends up

**Table 7.12.** Informativity for position and orientation in navigation plane. Process noise is chosen by naive tuning - S1.

| | $t^e_{ei}(1)$ | | | | $t^e_{ei}(3)$ | | | |
|---|---|---|---|---|---|---|---|---|
| | $\mathcal{I}_{0.5}$ | $\mathcal{I}_1$ | $\mathcal{I}_2$ | $\mathcal{I}_3$ | $\mathcal{I}_{0.5}$ | $\mathcal{I}_1$ | $\mathcal{I}_2$ | $\mathcal{I}_3$ |
| ES | −31 | −42 | −38 | −24 | −30 | −44 | −39 | −25 |
| ID | −24 | −38 | −45 | −17 | −20 | −37 | −46 | −30 |
| PD | −3 | −9 | −7 | 0 | −17 | −8 | 0 | 0 |

| | $\Phi^e_i(2)$ | | | |
|---|---|---|---|---|
| | $\mathcal{I}_{0.5}$ | $\mathcal{I}_1$ | $\mathcal{I}_2$ | $\mathcal{I}_3$ |
| ES | −37 | −64 | −87 | −87 |
| ID | −37 | −65 | −85 | −66 |
| PD | −24 | −27 | −12 | 0 |

with a multiple of the error in height, compared to ID and PD.

**Consistency** is one of the main focuses in this thesis. Although a test on a single trajectory cannot deliver meaningful statistical analysis, the quality of error appraisement can be evaluated. Thus, the percentage of classified inliers over all estimated poses was used to determine the informativity measure, which is given in table 7.12.

The deviations of observed inliers from the expected inlier count imply, that neither the ES nor the ID representations are able to provide reliable error bounds. However, as for the synthetic tests, the PD estimator determines a valid $3\sigma$-bound. For the lower bounds the results for the position and orientation have to be reviewed separately. Except for a few outliers the position estimates are consistent to the higher as well as the lower $\sigma$-bounds. In contrast to that, the orientation becomes inconsistent, in especially for the $\frac{1}{2}\sigma$- and $1\sigma$-bounds. Even though these are rarely used for error appraisement, filter runs of increased length might end up with an inconsistent orientation.

For visualization of error evolution the trajectory error and the respective $3\sigma$-bounds are given in figures 7.7, 7.8 and 7.9. It can be seen, that PD provides valid bounds for the estimation error. For ES, these bounds give

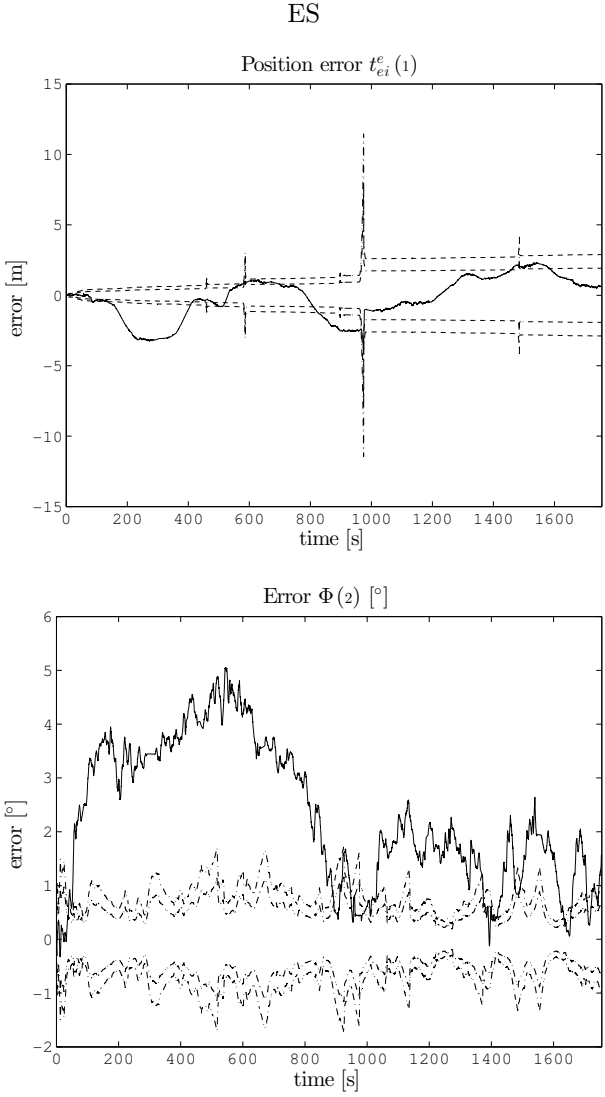


**Figure 7.7.** Estimation error (solid lines) and respective 2- and $3\sigma$-bounds for position in $\mathbf{e}_1$-axis and heading, i. e., plane navigation only. Parameter setup S1 (heuristic tuning).

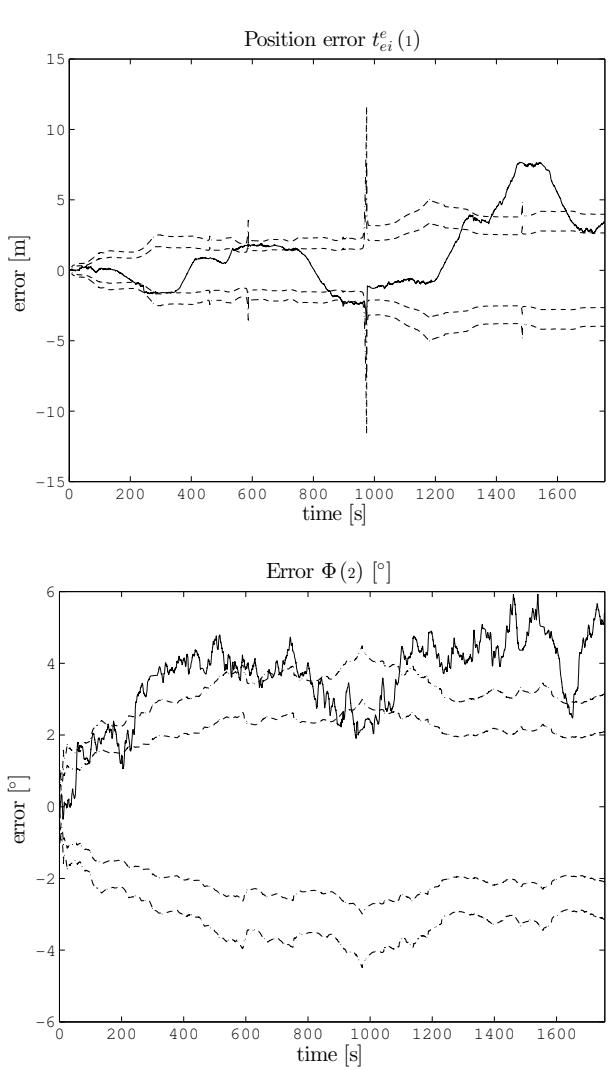


**Figure 7.8.** Estimation error (solid lines) and respective 2- and $3\sigma$-bounds for position in $\mathbf{e}_1$-axis and heading, i. e., plane navigation only. Parameter setup S1 (heuristic tuning).

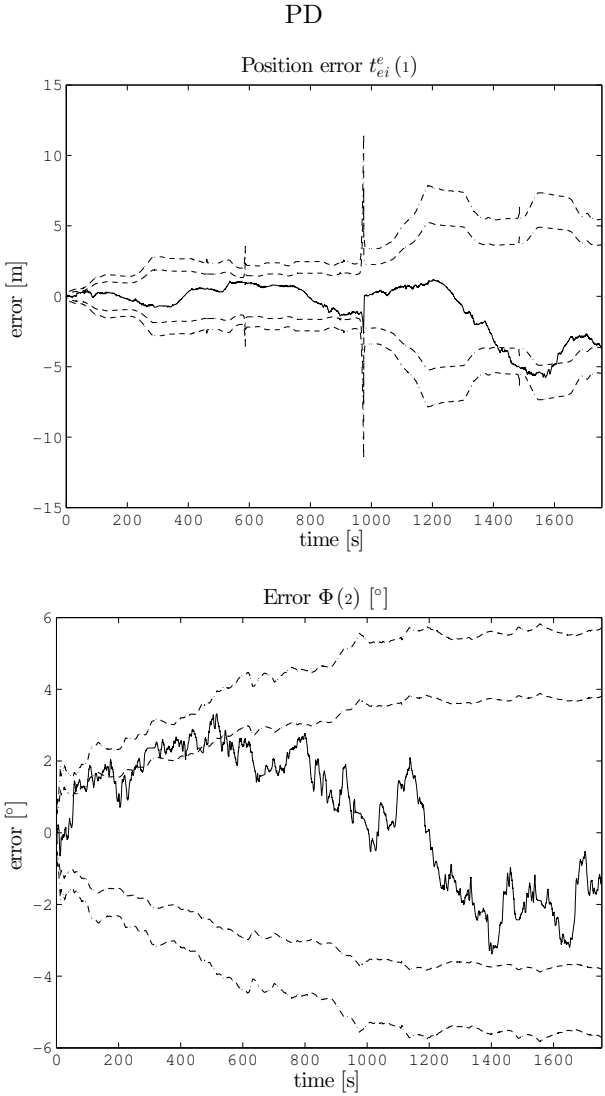


**Figure 7.9.** Estimation error (solid lines) and respective 2- and $3\sigma$-bounds for position in $\mathbf{e}_1$-axis and heading, i. e., plane navigation only. Parameter setup S1 (heuristic tuning).

no notion on the amount of error to be expected. Although the previous analysis asserts invalidity of the $\sigma$-bounds for ID, they give a better 'visual' impression of error evolution. Such subjective valuations cannot be used for reliable system analysis, but they provide an improvement compared to ES.

**System Recovery** Special attention has to be called for time index 960 seconds. For a time slice of about 15 seconds (i. e., 239 images) no visual features were available. Thus, the only trajectory information was given by strap down navigation provided by the IMU. In figure 7.10 the estimated trajectories in the $\mathsf{e}_1$-/$\mathsf{e}_3$-plane are visualized on the left, along with the respective lost-feature sub tracks on the right. The black marked tracks represent the images without landmark information.

Obviously the quality of strap down navigation rapidly decreases (in addition see figures 7.7, 7.8, 7.9). After detection of new landmarks, the system is able to correct the wrongly predicted velocity. Thanks to the consistent IMU prediction model, the correlations with the position and orientation are utilized by the Kalman Filter, to correct the failed prediction. Almost the complete pose error of 5 meters is compensated within 20 to 30 update steps. Moreover, the strongly increased state uncertainty is recovered up to the amount of not compensated drift (about 1 meter). Note that no re-detection of landmarks has been carried out. The pose correction was achieved solely by velocity to pose correlations for all parametrizations.

Such implicit pose corrections cannot be achieved using a constant velocity filter, since the velocity as well as the orientation changed during the non update timeslice. That is, during the lost tracking (see figure 7.10) the system stops in the corner (see ground truth trajectory) and turns it's view to the right heading floor. After this rotation the visual tracking was restarted.

This also proves the usefulness of Kalman Filters for IMU-aided SLAM. Bundle-adjustment and similar estimation techniques are not able to connect the two floors. This is to the fact, that no visual features are present in the floors corner. Thus, no visual connection can be made between them. For such estimators, the IMU prediction cannot be used for correlation based corrections. This is due to the fact, that no dynamic state model is used, and thus no time correlations can be exploited.

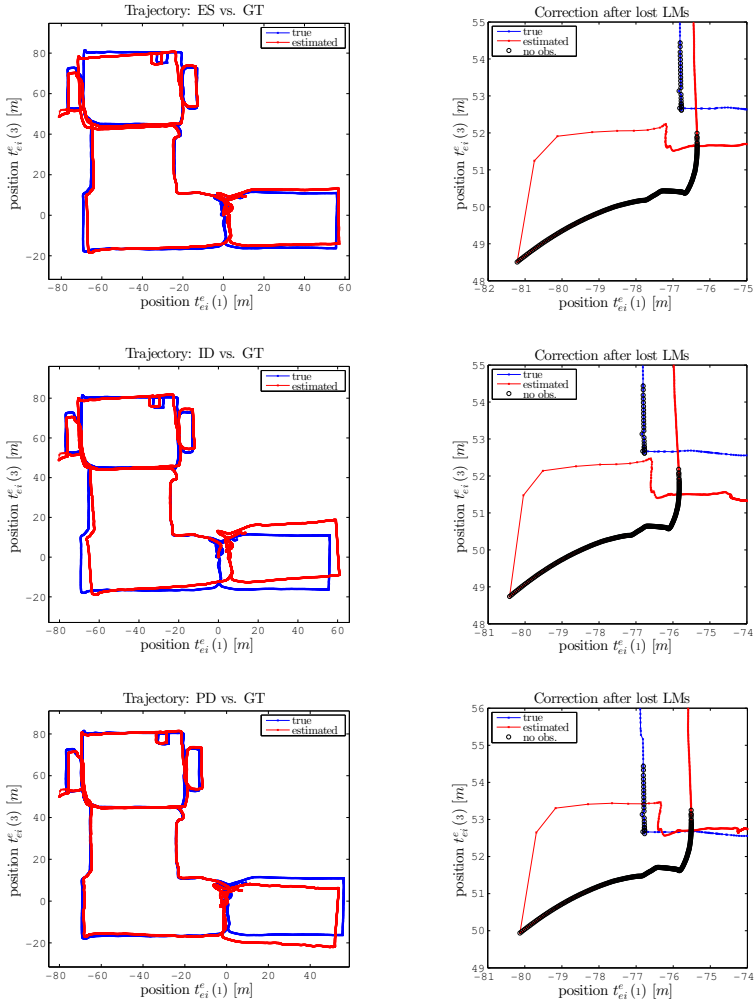


**Figure 7.10.** Trajectory estimation. On the left: estimated trajectory for parameter set S1 (heuristic tuning) using ES, ID and PD respectively. On the right in black: sub track of left side loop in plane E2 where no visual features were present (239 images $\hat{=}$ 15 seconds).

#### Integral Noise

The heuristic tuning of process noise parameters can be avoided, when precise knowledge of the errors in prediction modeling is available. In the following it is assumed, that the inertial movement model used for prediction (see equations (4.2.20) and (4.2.21), pages 80 and 80) is correct, i. e., no earth rotation and other interferences are present. Thus, the remaining prediction error is caused by the assumption, that the acceleration and angular velocity are constant for an integration step. Following section 4.3.2 (page 96), a prediction noise can be deduced using information on the expected change $\frac{\partial a}{\partial \tau}$ in acceleration and $\frac{\partial w}{\partial \tau}$ in angular velocity respectively.

To determine expectations for $\frac{\partial a}{\partial \tau}$ and $\frac{\partial w}{\partial \tau}$, IMU data for the RawSeeds dataset and free-hand movements (see section 7.3) have been analyzed. For these tests the values of $\frac{\partial a}{\partial \tau} \approx 2\left[\frac{m}{s^3}\right]$ and $\frac{\partial w}{\partial \tau} \approx 29\left[\frac{\circ}{s^2}\right]$ turned out. Applying equations (4.3.8) from page 93 for a sample rate of $125\left[\frac{1}{s}\right]$ results an expectation for the absolute integral error of about

$$\epsilon_t \approx 2{\cdot}10^{-7}\,[m] \qquad \epsilon_v \approx 6{\cdot}10^{-5}\left[\tfrac{m}{s}\right] \qquad \epsilon_\Phi \approx 1{\cdot}10^{-3}\,[^\circ] \approx 2{\cdot}10^{-5}\,[rad]$$

per sample for the position, velocity and orientation respectively. These values are used to approximate the standard deviations for the poses predictions process noise. In the following S2 refers to this parameter setup.

**Estimation Quality** In table 7.13 the resulting errors for S2 are given. Comparing this to the results for S1 (table 7.10), the RMS and AT errors for ES increased significantly. Decreasing the process noise the inconsistencies, observed for S1, are amplified. By this, the estimator increases the weight for the (previously too strong weighted) navigation prediction, resulting in a less informative estimation (compare tables 7.12 and 7.14). This self amplifying process degrades the estimation and possibly leads to filter divergence.

For the ID estimator the estimation quality remains at a similar level. As was shown in section 7.1.2, the landmark estimation for the ID representation is consistent, providing appropriate linearization scenarios are given. Due to the high prediction and update rate this can be assumed to be valid for this trajectory, since no extreme changes in the landmarks view direction occur. Although the poses informativity measures for ES and ID were similar in setup S1, the full filter state for ID is thereby more informative in comparison

**Table 7.13.** Plane navigation error analysis for S2 (integral process noise). Root mean square errors, absolute trajectory ATE and relative pose error RPE for the estimated trajectory over all 26335 poses.

| | RMSE | | |
|---|---|---|---|
| | $t^e_{ei(1)}$ $[m]$ | $t^e_{ei(3)}$ $[m]$ | $\Phi^e_{i(2)}$ $[^\circ]$ |
| ES | 2.675 | 2.099 | 2.951 |
| ID | 2.626 | 2.782 | 3.998 |
| PD | 1.479 | 0.962 | 2.279 |

| | Rawseeds error | | |
|---|---|---|---|
| | ATE $\bar{\varepsilon} \pm \sigma_\varepsilon$ | T-RPE $[m]$ | R-RPE $[^\circ]$ |
| ES | 3.0994 $\pm$ 1.3997 | 0.5173 | 0.9045 |
| ID | 3.1389 $\pm$ 2.1879 | 0.7407 | 1.07 |
| PD | 1.4773 $\pm$ 0.9652 | 0.8568 | 0.9672 |

**Table 7.14.** Informativity for position and orientation in navigation plane. Process noise is chosen by approximations of integration error - S2.

| | $t^e_{ei(1)}$ | | | | $t^e_{ei(3)}$ | | | |
|---|---|---|---|---|---|---|---|---|
| | $\mathcal{I}_{0.5}$ | $\mathcal{I}_1$ | $\mathcal{I}_2$ | $\mathcal{I}_3$ | $\mathcal{I}_{0.5}$ | $\mathcal{I}_1$ | $\mathcal{I}_2$ | $\mathcal{I}_3$ |
| ES | $-37$ | $-62$ | $-83$ | $-76$ | $-37$ | $-58$ | $-61$ | $-54$ |
| ID | $-28$ | $-39$ | $-44$ | $-32$ | $-17$ | $-35$ | $-44$ | $-32$ |
| PD | $-16$ | $-18$ | $-6$ | $-8$ | $-3$ | 4 | $-5$ | $-2$ |

| | $\Phi^e_{i(2)}$ | | | |
|---|---|---|---|---|
| | $\mathcal{I}_{0.5}$ | $\mathcal{I}_1$ | $\mathcal{I}_2$ | $\mathcal{I}_3$ |
| ES | $-38$ | $-66$ | $-92$ | $-94$ |
| ID | $-37$ | $-66$ | $-92$ | $-88$ |
| PD | $-25$ | $-33$ | $-42$ | $-25$ |

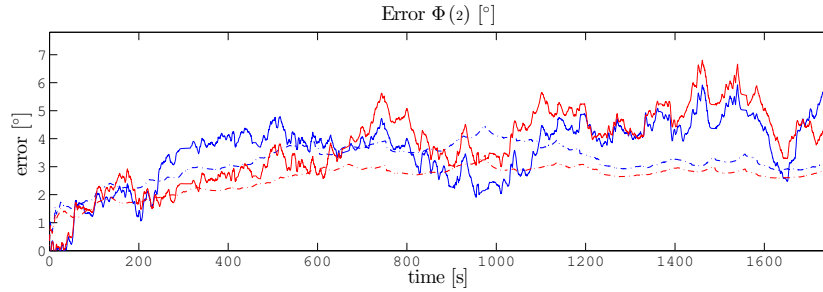


**Figure 7.11.** Comparison of ID heading error. Blue S1, red S2, dotted lines $3\sigma$-bounds.

to ES. Thus, decreasing the process noise has a reduced impact to ID. This also holds for the informativity measures in tables 7.12 and 7.14, which have not been effected significantly. Nevertheless, a slight improvement of about 14% for the position component $t^e_{ei}$ can be observed. Since the error evolution in heading changes it's characteristics slighlty (see figure 7.11), this is more likely to be a side effect than an estimation improvement. This is supported by the fact, that the estimated trajectories do not change their shape in contrast to PD (see figure B.6, page 239 for comparison).

For the PD representation the position error is significantly decreased. From the informativity measures in table 7.14 it can be seen, that the position estimation still is informative. In figure 7.12 the error in estimation and $3\sigma$-bound for position $t^e_{ei(3)}$ of the PD estimator is depicted. As can be seen, the estimated $3\sigma$ error bounds for S2 (marked red) are decreased but valid. By this, more of the update information is shifted to the IMU nuisance parameters, improving their quality (see the improved convergency rate for the acceleration bias in figure 7.13). This induces an improved position estimation.

In contrast to the position the error in heading is increased. As was mentioned in section 4.3.1 (page 95) the earths turn rate is neglected in the used movement model. For the S1 setup the process noise was sufficient to compensate for this model error. Reducing the predictions uncertainty to the pure integration noise results in an inconsistent heading estimation

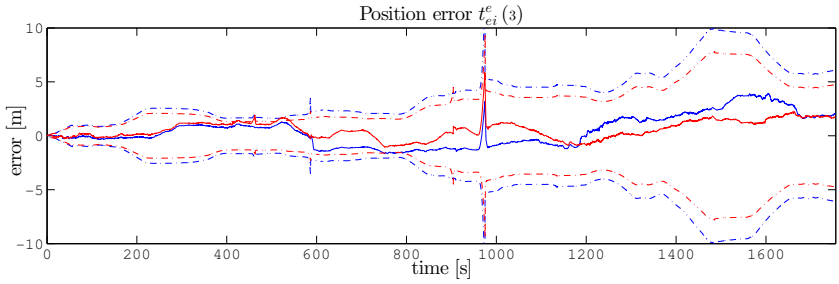


**Figure 7.12.** Comparison of PD position error $t^e_{ei(3)}$. Blue S1, red S2, dotted lines $3\sigma$-bounds.

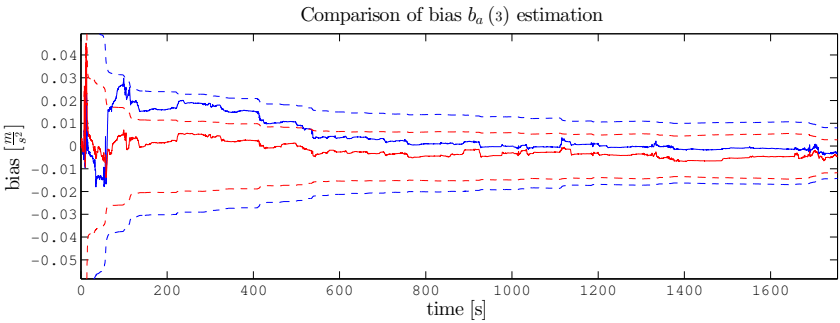


**Figure 7.13.** Comparison of PD acceleration bias $b_{a(3)}$ estimations. Blue S1, red S2, dotted lines $3\sigma$-bounds.

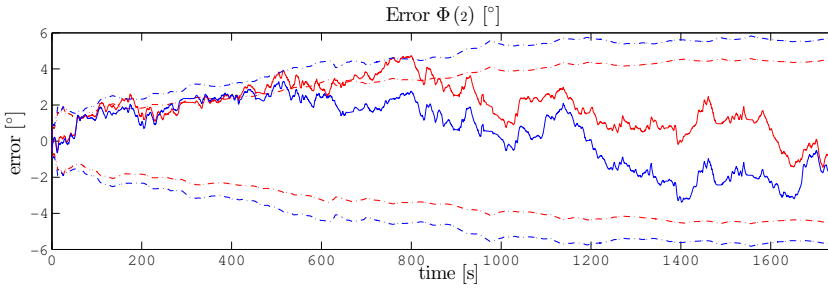


**Figure 7.14.** Comparison of PD heading error. Blue S1, red S2, dotted lines $3\sigma$-bounds.

**Table 7.15.** Error analysis in final height $t^e_{ei(2)}$ for S2 (integral process noise). $\epsilon$ final error, $3\sigma$ confidence interval and RMSE for floor E1.

| | $\epsilon$ | $3\sigma$ | RMSE |
|---|---|---|---|
| ES | -8.012 | $\pm$ 0.603 | 6.054 |
| ID | -1.701 | $\pm$ 1.246 | 1.129 |
| PD | -1.810 | $\pm$ 1.285 | 1.291 |

due to model violations. In figure 7.14 the heading estimation for both parameter setups are visualized. As can be seen, for S1 the heading is estimated consistently with respect to the $3\sigma$-bounds. In contrast to that S2s heading exceeds the bound after 100 seconds of estimation. Although the error characteristics for S1 and S2 are quite similar, the PD estimator is not able to correct the heading error for S2 in the same way as for S1. The heading error starts decreasing after time index 500 seconds. This is delayed by 300 seconds for S2. This is due to the fact, that the decreased uncertainties increase the low pass filter characteristics of the KF for long term orientation estimation.

For the estimation of the height $t^e_{ei(2)}$ results for S2 can be observed, that are similar to S1. In table 7.15 it can be seen, that the ES estimator is unable to determine a valid height. It is highly inconsistent and erroneous. In contrast to that, ID and PD perform similar for the height as for the

remaining position components. Nevertheless, as for S1, the height is the only inconsistent position estimation in PD.

#### Computational Effort

To validate the proposed systems applicability to real-time demands, the amount of runtime, needed for the different processing steps, has to be analyzed. To do so, the filter process is partitioned into the phases remove, predict, update, reinit and insert. The amount of time, needed for visual landmark tracking, is not analyzed, since it is out of the scope of this thesis. The remove step is made up of the outlier detection (observation prediction and $\chi^2$-test, see section 6.5.2) and the landmark deletion. The gathering of IMU data, application of navigation models and the respective error propagation are combined in the predict phase. The update step corresponds to a Kalman Filter update including error propagation. The reinit process is made up of the normalization of incremental rotations (see section 3.3.4) and additionally the relocalization for PD. The final insert phase introduces newly detected stereo correspondences to the filter state and applies the respective error propagation.

To ensure an upper bound for the computational effort, all parametrizations are limited to use 20 landmarks for updates (see section 7.1.1, page 162). Moreover, due to the long runtime of landmark initializations in ES and ID these have been restricted to insert at most 20 landmarks per insertion step (at most 40 landmarks in filter state). For PD the latter limit was not active. Thanks to the negligible runtime for PD initialization an arbitrary number of landmarks can be inserted (maximum of 422 for S1).

In figure 7.15, the timings in seconds for the processing steps are visualized for test setup S1. Table 7.16 gives the respective average frame rates. As can be seen, the results for the overall runtime are similar to those discussed in section 7.1.1, page 162. Comparing the ES and the ID estimator the latter has an increased computational effort for every filter phase. This is due to the increased state (landmark) size for ID. Nevertheless, both estimators are suitable for most real-time applications. But peeks down to 6.6Hz for a mean frame-rate of 17Hz for ID are not acceptable for some application areas. Since the very most of the runtime is spend in feature initialization, reducing the maximal number of landmarks to insert would significantly

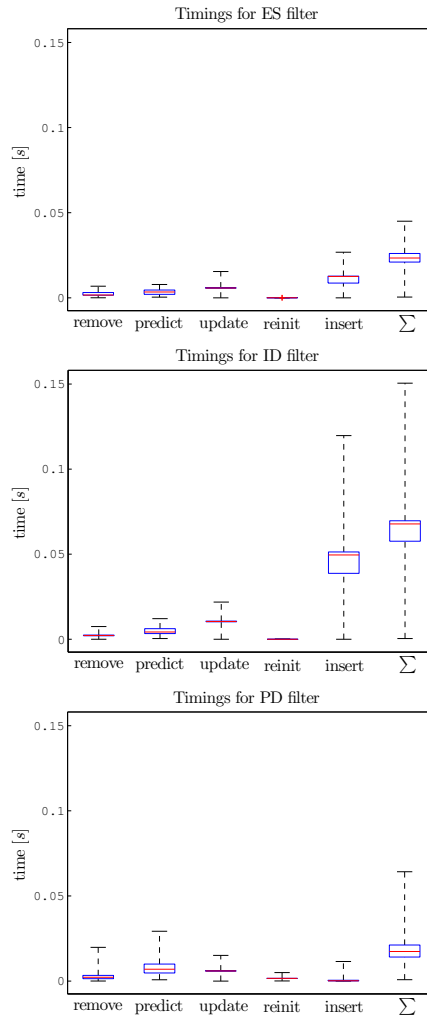


**Figure 7.15.** Visualization of runtime effort for parameter setup S1. Average number of landmarks in state/update, ES: 31.7/18.3; ID: 31.7/18.3; PD: 61.8/18.3. Lines mark minimum/maximum and red bars average values. Boxes give the 25% to 75% area (50% of measures inside box).

**Table 7.16.** Average frame rates for the estimator and the respective processing phases. Overall frame-rate is without landmark tracking.

| | remove | predict | update | reinit | insert | $\sum$ |
|---|---|---|---|---|---|---|
| ES | 449 | 299 | 181 | 86091 | 98 | 47 |
| ID | 455 | 207 | 101 | 64161 | 23 | 17 |
| PD | 396 | 134 | 182 | 684 | 2495 | 58 |

speed up the estimator. Although this goes along with a degraded quality of landmark tracking, this is acceptable for a strong decrease of runtime under certain circumstances.

The PD estimator has the highest average overall frame-rate. Nevertheless, the ES estimator is more stable with respect to the maximal runtime. I. e., the minimal frames per seconds are $> 20$Hz, where PD goes down to 16Hz. A noticeable fact is, that the remove and prediction phases need much more time in PD, compared to ES and ID. This is due to the fact, that the PD estimator does not limit the maximum number of landmarks for insertion. ES and ID states contain 40 landmarks at most, the PD states includes up to 422. Thus, the runtime for outlier detection (remove phase) and the error propagation for prediction is strongly increased. On the one hand, limiting the number of incorporated landmarks can speed up PD further, to reach minimal frame rates of $> 30$Hz. On the other hand, the large number of landmarks, that have passed the outlier test at least once, allow for building up a large landmark database.

It is worth mentioning, that allowing for an insertion of more landmarks, than are used for update, provides a high number of observed landmarks. For this test run in more than 23000 images ($> 87\%$ of updates) all possible 20 landmark slots were used. Due to the high number of active landmarks after pose prediction, up to 20 landmarks can be lost (outlier detection, failed tracking), leaving 20 in the filter state for update.

### 7.2.3 Camera to IMU Alignment

The full filter model includes the alignment between the camera and the IMU sensors. As stated before, the calibration method, introduced in

section 6.9, delivers high quality estimations for this calibration. Since no suitable calibration sequence is available for the RawSeeds dataset, the given calibration is used as initialization for online calibration. Since nothing is known about the reliability of this calibration, it is assumed to be uncertain up to a few centimeters and degrees respectively.

The estimated alignments $\Gamma_i^c$ for ES, ID and PD on setup S1 are visualized in figures B.7, B.8 and B.9 respectively (pages 240ff). As can be seen, ID and PD are able to determine a consistent calibration translation and rotation. Both parametrizations end up with a similar calibration. The only exception is the drift in estimation of the translation component $t^c_{ci(2)}$. Due to the planar trajectory this component corresponds to the systems height $t^e_{ei(2)}$, being the most unstable for all estimations (see section 7.2.2, page 175). From this it can be concluded, that the missing movement along the trajectories height and the missing horizontal rotation ($\Phi^e_{c(1)}$ and $\Phi^e_{c(3)}$) cause the error in the respective components not to be distinguishable for the KF.

In contrast to ID and PD, the ES estimator is not able to determine a valid rotational alignment of the camera and the IMU. Figure B.7 shows, that the estimations for the components $\Phi^c_{i(1:2)}$ are degenerated. I. e., they differ strongly from the ones for ID and PD and have a reduced uncertainty. This is due to the highly inconsistent estimation of the systems orientation. It's degenerated uncertainty causes the filter to wrongly distribute the information provided by the observations. Due to the planar movement the alignment $\Phi^c_{i(1:2)}$ corresponds to the systems horizontal orientation. This clarifies the fact, that the error in height $t^e_{ei(2)}$ estimation in ES is 4 times higher, compared to ID and PD (see table 7.11, page 175).

Additional tests were carried out, to determine the amount of orientational alignment, that can be compensated by the estimators. Therefore, the initial value for $\Phi^c_i$ angles was set to 6° with appropriate uncertainties. The ID and PD representations were able, to compensate these misalignment, as can be seen in figure 7.16. Even though ID is slightly inconsistent for the first 120 seconds (see right graph), they end up with the same estimate $\Phi^c_{i(1)} \approx -0.5°$ as for the standard S1 test. In contrast to this, ES does not compensate the initial misalignment. The initial value of 6° isn't changed significantly (to $\Phi^i_{c(1)} = 5.74°$, $\Phi^i_{c(2)} = 5.76°$), due to the highly inconsistent covariance (see degenerated $3\sigma$-bound in right graph). Thus, the ES

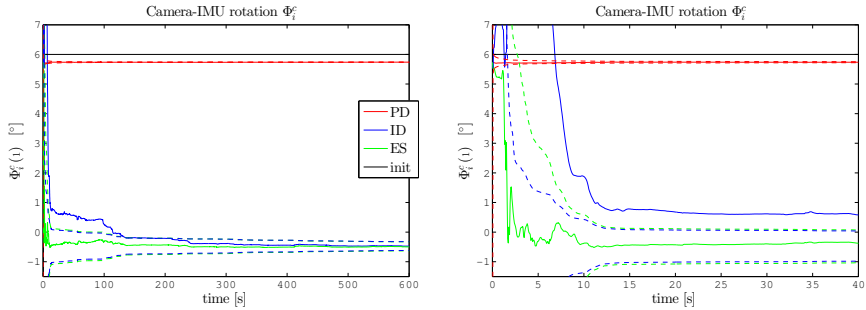


**Figure 7.16.** Comparison of camera to IMU calibration estimation for increased initialization errors. Exemplary rotation component $\Phi_{(1)}$. Right: cutout for first 40 seconds.

parametrization is not applicable to the IMU-camera calibration models.

Due to this misalignment the observed error in the estimation of $t^e_{ei(2)}$ (systems height) increases unbounded. As can be seen in table C.7 (page 248), ID and PD end up with the same final height error of 1 to 2 meters. The ES estimator undergoes a constant unbounded drift resulting in a final error of 80 meters ($> 10\%$ of trajectory length). That is 10 times more compared to the low $\Phi^i_c$ initialization error and 60 to 70 times more than ID and PD. This can be concluded by comparing the trajectory estimation errors of the standard test setup S1 and the one with increased initialization error (see table C.6, page 248).

**Preceeding Alignment** Increasing the initialization error further (i. e., to $> 10°$), introduces a high prediction error during startup. Moreover, due the calibrations high uncertainties (needed for consistent initialization), the predicted poses become highly uncertain. Depending on the movement at system startup this can end up in filter divergency for all parametrizations. This is because of the degraded linearization of high uncertain orientations. This leads to the conclusion, that the calibration tool, described in section 6.9, is recommended for close to unknown camera to IMU alignments. Nevertheless, for misalignments known with a precision of up to a few degrees, the auto calibration in the proposed SLAM system is sufficient.

**Table 7.17.** Plane navigation error analysis for PD with included image undistortion modeling for parameter setups S1 and S2. Root mean square errors, absolute trajectory ATE and relative pose error RPE for the estimated trajectory over all 26335 poses. Second row for each parameter setup gives RMSEs in percentage of estimation without undistortion models.

| Set | RMSE | | |
|---|---|---|---|
| | $t^e_{ei(1)}$ $[m]$ | $t^e_{ei(3)}$ $[m]$ | $\Phi^e_{i(2)}$ $[°]$ |
| S1 | 1.655 | 1.100 | 1.796 |
| | 79% | 71% | 96% |
| S2 | 1.528 | 0.938 | 2.655 |
| | 104% | 98% | 116% |

| Set | Rawseeds error | | |
|---|---|---|---|
| | ATE $\bar{\varepsilon} \pm \sigma_\varepsilon$ | T-RPE $[m]$ | R-RPE $[°]$ |
| S1 | 1.5917 $\pm$ 1.1907 | 0.5754 | 0.9030 |
| | 82% | 86% | 106% |
| S2 | 1.4936 $\pm$ 0.9921 | 0.9662 | 1.1028 |
| | 101% | 113% | 114% |

### 7.2.4 Image Undistortion Model

In the following the influence of modeling the image normalization mapping $h_n$ within the observation models is analyzed. To do so, additional tests were carried out using the method described in section 4.3.1 on page 93. For evaluation of the models impact the PD estimation for the S2 setup is applied to the augmented observation model.

In table 7.17 the results of the PD estimation without (see section 7.2.2, page 174) and with the pixel normalization model are compared. As can be seen, the deviations between both methods are irregular. On the one hand, the RMSE for position estimation in setup S1 decreases noticeably. On the other hand, the rotation and relative errors for setup S2 are increased.

For the informativity measures, given in table 7.18 similar results apply. Although some slight improvements occur, the overall performance is not improved significantly. It is especially noticeable, that for the orientation in S2 the consistency is improved, although it's RMSE is increased. This

**Table 7.18.** Informativity of PD for position and orientation in navigation plane with included image undistortion model. Parameter setups are compared to the estimations without undistortion model (according to tables 7.12, 7.14).

| | | $t^e_{ei}(1)$ | | | | $t^e_{ei}(3)$ | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | $\mathcal{I}_{0.5}$ | $\mathcal{I}_1$ | $\mathcal{I}_2$ | $\mathcal{I}_3$ | $\mathcal{I}_{0.5}$ | $\mathcal{I}_1$ | $\mathcal{I}_2$ | $\mathcal{I}_3$ |
| S1 | with | −2 | −7 | 1 | 0 | 3 | 2 | −1 | 0 |
| | wo. | −3 | −9 | −7 | 0 | −17 | −8 | 0 | 0 |
| S2 | with | −4 | 0 | −5 | −7 | 13 | 12 | −1 | 0 |
| | wo. | −16 | −18 | −6 | −8 | −3 | 4 | −5 | −2 |

| | | $\Phi^e_i(2)$ | | | |
|---|---|---|---|---|---|
| | | $\mathcal{I}_{0.5}$ | $\mathcal{I}_1$ | $\mathcal{I}_2$ | $\mathcal{I}_3$ |
| S1 | with | −14 | −21 | −13 | 0 |
| | wo. | −24 | −27 | −12 | 0 |
| S2 | with | −20 | −27 | −27 | −10 |
| | wo. | −25 | −33 | −42 | −25 |

**Table 7.19.** Final height $t^e_{ei}(2)$ error analysis for PD estimation including image undistortion model. $\epsilon$ final error, $3\sigma$ confidence interval for final estimation and RMSE for floor E1. Comparison to tables 7.11 and 7.15.

| S1 | | | S2 | | |
|---|---|---|---|---|---|
| $\epsilon$ | $3\sigma$ | RMSE | $\epsilon$ | $3\sigma$ | RMSE |
| -1.985 | ± 1.435 | 1.383 | -2.007 | ± 1.409 | 1.477 |
| 113% | NA | 114% | 111% | NA | 114% |

leads to the conclusion, that the augmented observation model has a certain impact on the covariance evolution. Nevertheless, the correlations cannot be considered as more close to the true ones, since the estimation error is increased.

For the final comparison table 7.19 depicts the errors in height $t^e_{ei}(2)$ estimation. Since the $3\sigma$-bounds are dedicated to $\epsilon$ for the respective estimation, they are not related to the previous estimations. As can be

seen, the quality for the estimations with undistortion model are degraded compared to the ones without augmented modeling. Moreover, the final deviations $\epsilon$ are inconsistent at the same amount, as they are for the standard models.

Concluding it can be stated, that the inclusion of undistortion models within the filter update does not improve the estimation quality. Moreover, the computational effort increases by 5% to 10%, depending on the state size. Thus, modeling the observations covariance distortion is not recommendable for systems with moderate lens distortion, as used in these tests.

### 7.2.5 Adjustment

The post-processing of estimated trajectories is a powerful tool, to improve the estimations for localization and mapping. In especially when loop closes or absolute references for sub tracks are known, the reconstructions quality can be increased significantly. This is done using an adjustment minimizing the overall error by exploiting additional information. As discussed in section 4.2.8 a simple adjustment constraint can be used for PD representations.

In figure 7.17 the positions of the selected loop closes are visualized. Since an automatic detection of loop closes is out of the scope of this thesis, they are selected manually. The relative pose between the loop close frames is approximated by interpolation from the datasets ground truth information. The interpolation is necessary since the ground truth contains only a subset of 6580 poses for the 26335 images.

Table 7.20 gives the RMSEs of the adjusted trajectory, visualized in figure 7.18. As can be seen, the estimation error is significantly improved. The position error was reduced to about 30%. Even though the rotational error is reduced only by 17%, the improvement is also appreciable. Especially noticeable is the improvement of sub tracks where no visual features were detected. Figure 7.19 visualizes a comparison of the filter inherent pose recovery due to the inertial motion model and the additional improvement by adjustment. As can be seen, the erroneous spatial prediction due to small errors in the determined orientation (gravity interferes with measured acceleration) can be compensated accurately. The state internal correlations have been exploited by the filter for recovery after updates succeeding the

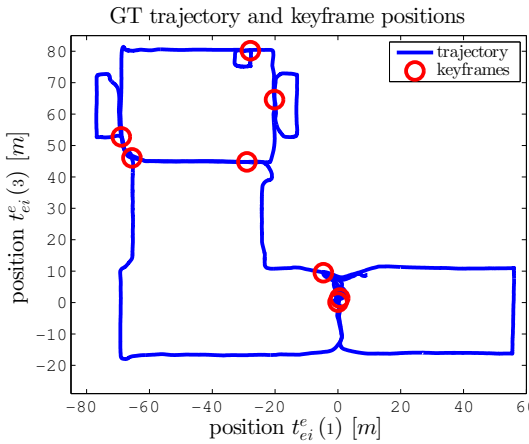


**Figure 7.17.** Position of the 8 selected loop closes for the Rawseeds dataset. The sizes of the loops range from 1700 to 25600 poses.

**Table 7.20.** Errors for the adjusted trajectory, test setup S1. Additionally the errors are given in percent of the estimation before adjustment.

| | RMSE | | |
|---|---|---|---|
| | $t^e_{ei}(1)$ $[m]$ | $t^e_{ei}(3)$ $[m]$ | $\Phi^e_i(2)$ $[°]$ |
| PD abs. | 0.678 $[m]$ | 0.394 $[m]$ | 1.55 $[°]$ |
| PD rel. | 32% | 25% | 83% |

| | Rawseeds error | | |
|---|---|---|---|
| | ATE $\bar{\varepsilon} \pm \sigma_\varepsilon$ | T-RPE | R-RPE |
| PD abs. | 0.661 $[m]$ $\pm$ 0.422 $[m]$ | 0.173 $[m]$ | 0.97 $[°]$ |
| PD rel. | 34% $\pm$ 24% | 26% | 114% |

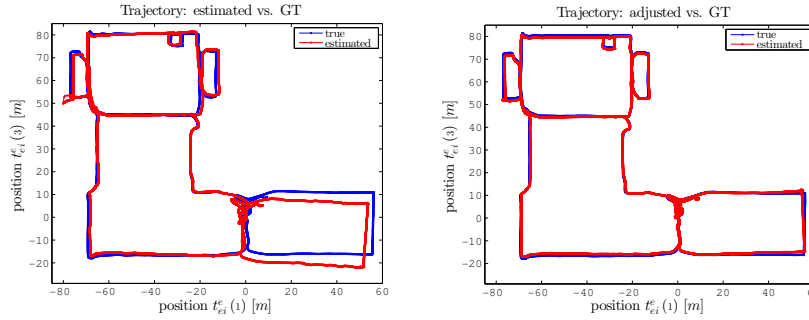


**Figure 7.18.** Adjusted trajectory path and ground truth information for setup S1 on the right. For comparison the left plot visualizes the trajectory before adjustment (see section 7.10, page 181).

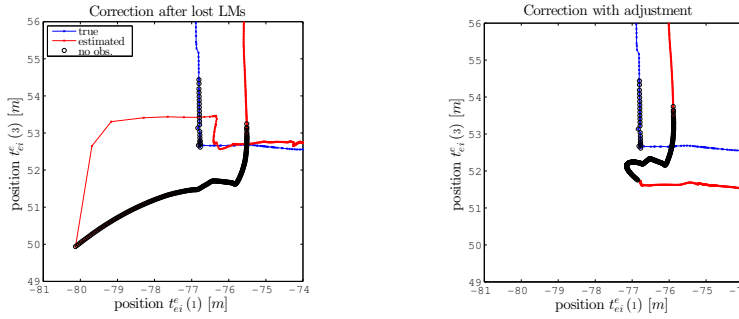


**Figure 7.19.** Pose recovery after images without visual features. The left trajectory depicts the filter estimation (same as PD trajectory in figure 7.10, page 181). The right trajectory visualizes additional improvement of not observed poses by application of adjustment.

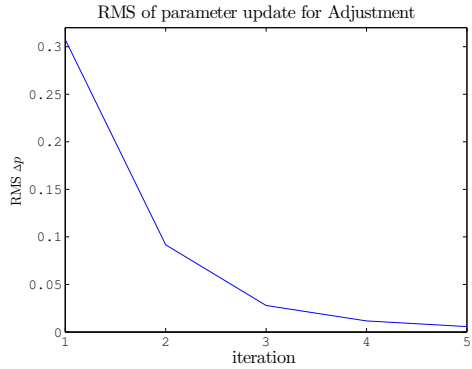


**Figure 7.20.** Root Mean Square of parameter update $\Delta p$ during adjustment using all 26335 poses and 8 loop closes.

not observed path. The same correlations are used in the adjustment model, such that the filter recovery can also be related to the poses preceeding this recovery. By this, the quality of the not observed poses is increased significantly.

For the non linear constraints (4.2.50) and (4.2.51) (see page 88) the iterated least squares estimator, defined in section 3.2, is used. Due to the small residual size, the only limiting factor for the computational effort is the creation of Jacobian matrices. For a number of $\mathtt{n}$ poses, the 6 pose components (position and orientation) and $\mathtt{m}$ loops the size of the constraints $g$ (see section 4.2.8, page 89) Jacobian is $6\mathtt{m} \times 6\mathtt{n}$. Thus, the system matrix, to be inverted for the WLSE, is only $6\mathtt{m} \times 6\mathtt{m}$. For the 8 loops used for adjustment in the following a system matrix of size $48 \times 48$ results. For the number of $\sim$ 26335 used poses, the adjustment has a memory consumption of less than 500 Megabytes.

Beside the memory consumption, the runtime effort is low enough, to allow for a loop close online and in parallel to the SLAM process. In figure 7.20 the magnitude of the parameter update is depicted in dependency of the number of performed estimation iterations. As can be seen, after 3 to 4 iterations the update vector is close to zero and the iteration can be terminated. This corresponds to a computing time of 15.3 to 20.4 minutes

for all 26335 involved poses. Since the respective trajectory was about 30 minutes, an online application of this adjustment is possible. Since the matrix multiplication (Jacobian and covariance) has block matrix structure, it can be computed efficiently on the fly. The needed runtime for the inversion of the system matrix of size $48 \times 48$ is negligible. Thus, by building up the Jacobian matrix during system runtime the first estimation iteration can be applied within a few seconds. Nevertheless, for the following iterations the Jacobian matrix has to be build up from scratch. Thus, they will take incomparable more time ($\sim$ 5 min. per iteration). Loop closes on sub tracks will decrease the computational effort proportional to the change in the number of poses, used for the constraint $g$.

### 7.2.6 Environment Mapping

The creation of environment maps is a basic task in SLAM. The PD representation provides a high number of landmarks due to the extensive landmark initialization. For the test setup S1, a total of 1,621,223 landmarks has been created. Since most of them have not been part of an update step, only a small part is useful for visualization. Nevertheless, all landmarks can be used for building a stochastic environment map. This can be accomplished using space occupancy probabilities, as is discussed in [GFP08] by Guan et al.

An advantage of the PD representation is, that the landmarks are parametrized using the local pose. Thus, after the above discussed adjustment the 3-space position of all landmarks can be re-computed. Due to the simple per landmark triangulation (see section 6.8.2, page 146) the respective computational effort is negligible. In figure 7.21 a simple point cloud representation of the environment map is visualized. The only post processing applied is the re-computation for the adjusted pose. To improve the visual appearance, the map was projected to the $\mathsf{e}_1$-/$\mathsf{e}_2$-plane and smoothed using a kernel density estimation (see [Par62], [DH03]). Moreover, only features being part of at least one update were used. This way a total of 481460 landmarks has been used.

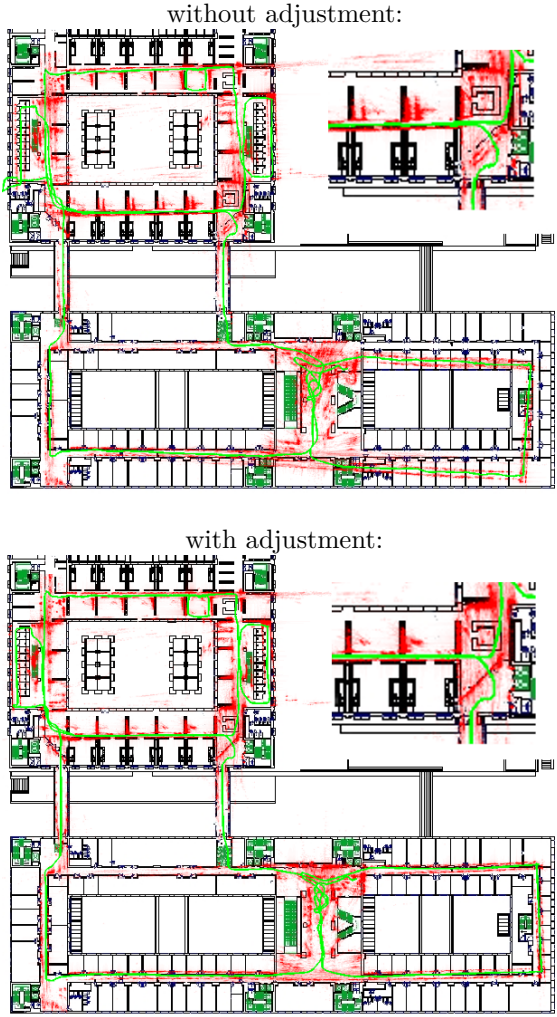


**Figure 7.21.** Visualization of reconstructed environment map and estimated trajectory. Top: without adjustment; Bottom: after adjustment (3 iterations < 16 min). Building floor plan provided along with Rawseeds dataset.

## 7.3 Free-Hand Tests

This section validates the real-time capability and robustness of the proposed SLAM system. This is done using "free-hand" tests in a full 6 DoF movement for the system being carried in an office environment.

Since no ground truth information is available for the traveled trajectories, each test starts and ends at the same position and orientation. This way the final error and it's consistency can be analyzed. For the PD parametrization three test scenarios were used. For the first two tests a dataset has been recorded, to allow for a direct comparison of the three parametrizations ES, ID and PD by using the same input data. The tests are performed for low and high resolution, i. e., $320 \times 240$ and $640 \times 480$ pixels. The final test for PD is performed online, using an image resolution of $320 \times 240$, to meet the maximum camera frame-rate.

For all experiments the same filter tuning is used. It was deduced using the results from the offline tests. Since the IMU for both systems have the similar hardware specifications, the respective values were used unchanged (see setup S1 in section 7.2.2, page 174).

### 7.3.1 Hardware and Environment

The hardware setup is made up of a Bumblebee XB3 (©by Point Grey Research GmbH) camera combination and a Xsens MTx (©by Xsens Technologies) inertial sensor. The XB3 provides a triplet of color images at 16Hz being synchronized in time. The used stereo images are taken from the outer two cameras resulting in a baseline of approximately $24\,cm$. For the tests image resolutions of $640 \times 480$ and $320 \times 240$ were used for visual processing. The latter limitation is due to the fact, that the bottleneck of the runtime performance is the feature tracking. For the higher resolution feature detection and landmark matching on both images are only available up to 10Hz. For the reduced resolution this step can be performed with $\geqslant$20Hz. By this, the filter exploits all images, that are provided by the cameras at 16Hz.

The IMU is rigidly mounted on the center of the camera housing. See figure 7.22 for a visualization of this setup. The camera to IMU calibration method (see section 6.9) has been used, to determine the respective

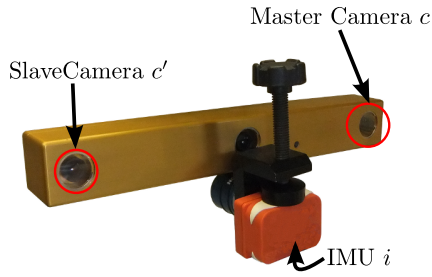


**Figure 7.22.** Hardware setup used for "free-hand" tests, Bumblebee XB3 (©by Point Grey Research GmbH) camera combination and the Xsens MTx (©by Xsens Technologies) IMU. Stereo images are taken from the outer cameras.

calibration $\Gamma_c^i$. To avoid the need of re-calibration, $\Gamma_c^i$ is used as initialization for the respective filter state components. Thus, small changes in the calibration due to imperfect storage and the like can be compensated. The alignment between the camera and the IMU is assumed, to not deviate from the determined calibration by more than a few degrees.

### 7.3.2 Recorded Dataset

The recorded dataset has a total length of 966 images ($60[s]$, $33[m]$). Although this is a short test sequence, it provides the opportunity, to compare the different estimators for free-hand trajectories. For the low image resolution the system is capable to estimate the pose and environment map with the full frame-rate of 16Hz. For the high resolution the bottleneck of image processing slows down the frame-rate to 8-11Hz for all parametrizations (keep in mind the unlimited landmark initialization in PD). Thus, processing all images cannot be done online. Nevertheless, the high resolution test gives a notion on the systems quality of performance when using improved image processing.

**Close Loop Error** As stated above, a closed loop has been traveled, to allow for an evaluation of the final estimation. In table 7.21 the deviations between the first and the last poses are given. As can be seen, ID and PD perform

**Table 7.21.** Deviations and respective $3\sigma$-bounds between start and end pose for recorded dataset. Low resolution is $320 \times 240$, high resolution $640 \times 480$

| | res. | $t^e_{ei(1)}$ $[m]$ | $t^e_{ei(2)}$ $[m]$ | $t^e_{ei(3)}$ $[m]$ | $\Phi^e_{i(2)}$ $[^\circ]$ |
|---|---|---|---|---|---|
| ES | low | $-0.957 \pm 0.492$ | $0.263 \pm 0.2$ | $-0.094 \pm 0.32$ | $< 1 \pm 1.55$ |
| | high | $-0.168 \pm 0.18$ | $0.232 \pm 0.09$ | $-0.071 \pm 0.17$ | $< 1 \pm 1.21$ |
| ID | low | $-0.075 \pm 0.24$ | $-0.02 \pm 0.13$ | $-0.019 \pm 0.22$ | $1 \pm 2.84$ |
| | high | $-0.011 \pm 0.18$ | $-0.039 \pm 0.09$ | $-0.114 \pm 0.18$ | $1.1 \pm 2.5$ |
| PD | low | $-0.109 \pm 0.23$ | $0.03 \pm 0.13$ | $-0.02 \pm 0.22$ | $< 1 \pm 3$ |
| | high | $0.057 \pm 0.16$ | $-0.04 \pm 0.08$ | $-0.056 \pm 0.16$ | $< 1 \pm 2.5$ |

similar for the low resolution setup, delivering a consistent estimation of high precision. In contrast to that the ES estimator undergoes a much higher error and delivers inconsistent covariances even for this short sequence. For the high resolution setup the ID and PD estimators cannot improve the estimation precision significantly. This is due to the fact, that visual features are available with high density in low and high resolution (15 vs 18 average landmarks per update). Moreover, the color images allow for a high quality feature matching using the SIFT descriptors. Nevertheless, the number of initialized features in PD is tripled (18151 vs. 53535). In contrast, the ES estimator is able to improve it's estimation quality for $t^e_{ei(1)}$, although the determined covariances remain inconsistent. The improvement in the positions quality is due to the fact, that the decreased observation uncertainty increases the impact of landmark observations. Thus, the quality decrease due to inconsistency can be reduced for this short test sequence.

**Camera-IMU Alignment** As mentioned above, the alignment is initialized using the method proposed in section 6.9. Due to this, the deviation from this calibration is estimated in this test run. In figure 7.23 the estimation of the camera to IMU rotation's third component $\Phi^c_{i(3)}$ for the high resolution test is visualized. Although both estimators end up with the same calibration (ID $-4.09°$, PD $-4.16°$), ID delivers inconsistent covariances for the first 7 seconds (see also figure 7.16, section 7.2.3 for Rawseeds dataset). Comparing the $3\sigma$-bounds it is conspicuous, that the covariances for ID decrease faster than for PD. Since the system was at no motion for the first 2.5 seconds,

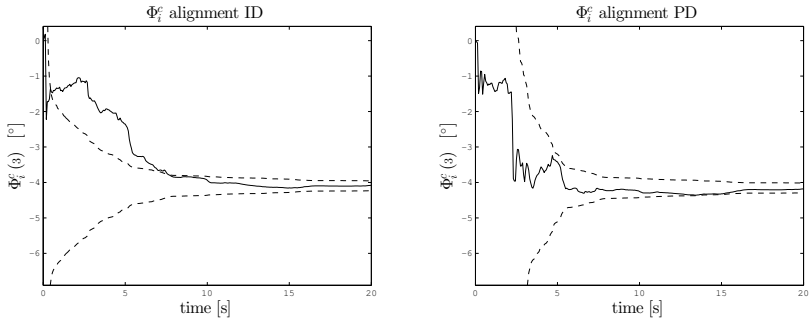


**Figure 7.23.** ID and PD estimations for the third component of the camera to IMU rotation $\Phi_i^c$ and respective $3\sigma$-bounds.

no information on the rotational alignment was available. Thus, the true covariances cannot decrease, what ID does not comply with. In contrast to that, PD correctly detects the non observable state component.

**Environment Map** The density of the estimated environment map for ES and ID differs strongly from the density for PD. For ES 13684 landmarks for the low and 14750 landmarks for the high resolution were created. The ID representation was able to increase the number of landmarks to 15380 and 16140 respectively. Thanks to the massive creation of stable landmarks (see implementation chapter 6) the size of the generated point cloud is increased, especially for the high resolution. That is, the environment maps contain 18151 and 53535 landmarks for low and high resolution respectively.

In figure 7.24 the estimated path and reconstructed environment map for the high resolution PD estimation are visualized. Additionally, an approximation of the respective floor plan is depicted. The only post processing, that has been applied to the map, is a KDE (as for section 7.2.6) for visual smoothing. The map is created using all 53535 created landmarks. As can be seen, the floor plan is matched and even miscellaneous objects, like the marked mobile blackboard and it's rack standing in front of the wall, can be detected as obstacles.

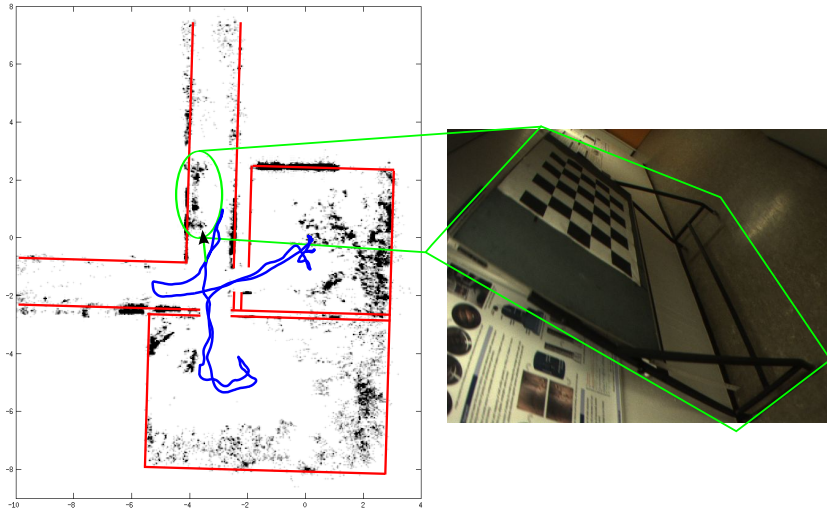

**Figure 7.24.** Reconstructed office map and trajectory (blue) using PD estimator for high resolution images (640 × 480). Red lines mark an approximative floor plan using perpendicular walls. Green ellipse marks mobile blackboard in front of the wall, arrow the respective camera view (see right image).

### 7.3.3 Online SLAM

This test is performed online, i. e., without any post processing, in the same office environment as in figure 7.24. It consists of 3200 images, corresponding to a period of time of 200 seconds. To allow for a more detailed evaluation of quality, the start pose has been revisited three times. The trajectory has a total length of 120 meters, traveled with an average speed of 0.6 meters per second and 40 degrees per second respectively. Moreover, spatial and angular accelerations of $> 3.5 \frac{m}{s^2}$ and $> 300 \frac{^\circ}{s^2}$ occur (computed from smoothed sensor data). In figure 7.25 the speed characteristics for the whole trajectory are depicted. The high variations in speed and turn rates can be assumed, to match those of a freely moving human being, i. e., a free-hand movement.

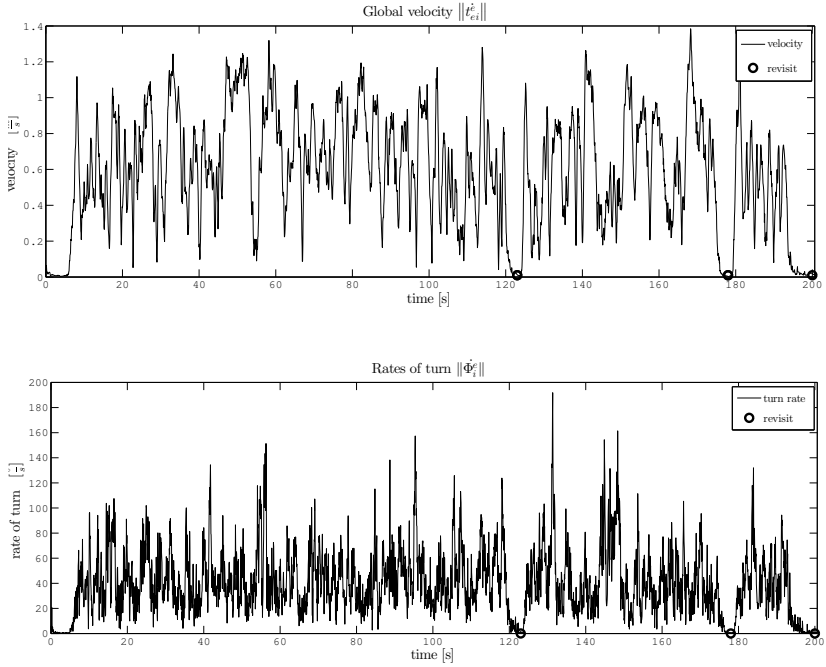


**Figure 7.25.** Movement characteristics for online free-hand test run. The circles mark revisits of initial system pose (loop closes). Environment is the same as in recorded dataset (see figure 7.24). Top: Estimated absolute velocities; Bottom: turn rates as derivatives of low-pass filtered (to reduce jitter) estimated orientation $\Phi_i^e$;

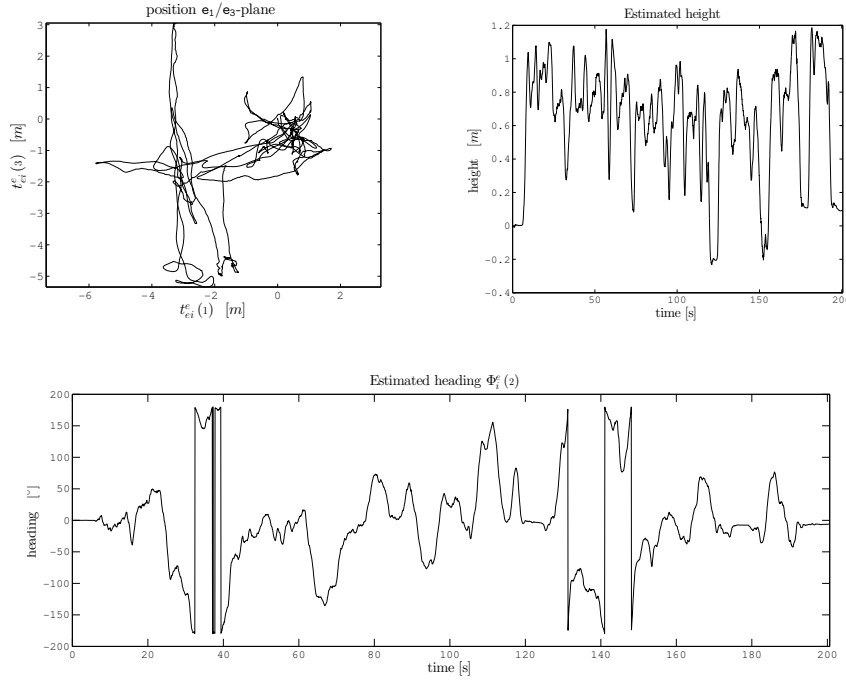


**Figure 7.26**. Estimated trajectory, height $-t^e_{ei}(2)$ and heading $\Phi^e_i(2)$ for online free-hand test run. Environment is the same as in recorded dataset (see figure 7.24).

**Close-Loop Error** In figure 7.26 the estimated trajectory ($\mathsf{e}_1/\mathsf{e}_3$-plane), the respective height profile and the horizontal orientation are visualized. To be able to evaluate the estimation quality, the start pose ($t^e_{ei} = 0_3$, $\Phi^e_i = 1^3$) has been revisited three times, i. e., loop closes at 122 and 180 seconds as well as at the trajectories end (see circle markers in figure 7.25). In table 7.22 the respective estimation errors and the according pose uncertainties are given. As can be seen, the pose estimation is consistent for all loop closes. The only exception is the heading at 180 seconds. Since it exceeds the $3\sigma$-bound by merely 13%, and meeting the initial pose is only precise up to $1° \sim 2°$, this can be assumed, to be within the statistical variations.

**Table 7.22.** Errors and respective $3\sigma$-bounds between start and revisited poses for the online test. The latter mark loop closes at 122, 180 seconds and the trajectories end.

| time $[s]$ | $t^e_{ei(1)}$ $[m]$ | $t^e_{ei(2)}$ $[m]$ | $t^e_{ei(3)}$ $[m]$ | $\Phi^e_{i(2)}$ $[^\circ]$ |
|---|---|---|---|---|
| 122 | $-0.550 \pm 0.63$ | $0.201 \pm 0.22$ | $-0.001 \pm 0.4$ | $-3.12 \pm 5.4$ |
| 180 | $0.258 \pm 0.71$ | $-0.108 \pm 0.36$ | $0.027 \pm 0.94$ | $-7.59 \pm 6.7$ |
| 200 | $0.415 \pm 0.76$ | $-0.094 \pm 0.39$ | $-0.491 \pm 0.99$ | $-5.88 \pm 6.6$ |

**Camera-IMU Alignment** As for the offline test scenario, the camera-IMU alignment has been determined consistently. This can be concluded from figure B.10 on page 243. The estimations for all alignment components converge to constant values. Moreover, the full estimation history stays consistent to the $3\sigma$-bounds, centered at the final value.

**Computational Effort** The computational effort in time for all processed frames did not exceed $\frac{1}{16}$ seconds for the complete SLAM system. That is, for the full camera frame-rate of 16Hz all captured images were handled in time. This includes the full visual tracking, filter management and estimation procedures as well as the environment mapping and data logging. It is worth noticing, that this holds for an unlimited initialization of landmarks. Thanks to the simple initialization process in PD parametrization a large environment map of 63732 landmark was created and handled in real-time.

## 7.4 Conclusion

The analysis of the discussed experiments showed the diverse fitness for real-time IMU-aided stereo SLAM for ES, ID and PD. This holds for the estimated trajectories and their covariances as well as for the determined camera-IMU calibration and the IMUs nuisance parameters.

**ES** The ES parametrizations applicability to stereo IMU SLAM is limited. For this representation the experimental analysis proved the inconsistent and non informative state estimation for synthetic as well as real world scenarios. These conclusions match the ones drawn in previous research (see [Bai+06],

[CNT04], [Cas+07], [HMR08]). Moreover, even small deviations in the initialization of the camera to IMU calibration result in a high estimation error. This reduces the estimation quality and can cause divergency due to the wrongly estimated camera to IMU rotation. Especially the test scenario using the Rawseeds dataset showed a constant and, depending on the initialization, high drift up to 10% of trajectory length in the system's height $t^e_{ei}(2)$. Moreover, the provided uncertainty valuation for all filter components is unserviceable. This clarifies the fact, that the IMUs nuisance parameters, the camera to IMU calibration and the system's height are highly erroneous and unreliable. Nevertheless, ES is capable, to determine a precise trajectory estimation for planar navigation in the given RawSeeds dataset.

**ID** In contrast to ES, the ID parametrization has proven the be far more informative. Moreover, it is more robust to erroneous calibration initialization. The calibration estimation is valid even for deviations in the rotation initialization up to several degrees. This also allows for a significantly reduced error for the systems height in comparison to ES. Nevertheless, for a precise system initialization the ID representation provides similar estimation errors for the 3DoF pose ($t^e_{ei}(1)$, $t^e_{ei}(3)$, $\Phi^e_i(2)$) in the planar navigation tests. On the other hand, ID is capable of determining trajectory information in the short free-hand test scenario in a far more precise and consistent way. To sum up, the ID parametrization is more suitable for stereo inertial aided SLAM compared to ES, in especially for full 6DoF movement. This agrees with the conclusions, drawn for vision only SLAM systems in previous research (see [CDM08], [Sol+12]).

**PD** The discussed synthetical and real world tests proved the validity of the newly introduced state parametrization for PD landmarks and it's applicability to actual SLAM platforms. The relative pose parametrization and camera local landmarks allow for a consistent estimation of all filter components. This holds especially for long term navigation, since the incremental navigations covariances are more robust to global drift. Moreover, the increments correspond to the movement prediction provided by IMUs. By this, the estimation of the camera to IMU calibration for PD is reliable and more consistent compared to ID. The consistency for the PD

estimation was ensured for all tests, except for moderate inconsistencies for the heading in RawSeeds test setup S2 due to the neglected earth rotation (see section 7.2.2, page 182). The online free-hand test leads to the conclusion, that the PD SLAM system is applicable to real-time tasks and it is capable of simultaneously estimating complex movements and large environment maps. Beside the improved runtime performance and map density with respect to ID, the availability of a simple adjustment requiring only low computational effort is a major advantage compared to ES and ID. It is capable of determining a trajectory estimation of high precision and an improved environment map without the need of bundle adjustment techniques.

Chapter 8

# Closing Remarks

As epilogue this chapter closes the thesis by providing an overview of it's contents and conclusions for the discussed analysis. For a final remark an outlook to future work is given.

## Summary

In this thesis the basics for visual SLAM algorithms and augmentations to IMU aided stereo SLAM have been discussed. In chapters 1 and 2 this thesis' task as well as the needed notations and foundations were introduced. Chapter 3 comprised an introduction to linear estimation. Moreover, the analysis tools for covariance consistency and linearization quality were introduced.

A detailed description of the referenced state of the art landmark parametrizations was given and the new PD model has been introduced in chapter 4. Initialization and observation models were derived, to solve the SLAM task and estimate an IMU to camera calibration simultaneously. Following this, a detailed analytical evaluation of the discussed models was performed in chapter 5.

Following the model design discussion a detailed description of the software framework, that has been used for evaluation in real world, was given in chapter 6. This software has been used, to validate the applicability of the discussed models to actual SLAM systems in chapter 7. Additionally the basic system models excluding the IMU to camera calibration were tested on synthetical data.

### Conclusions

To conclude this thesis, the results in the model evaluation chapter 5 and the experiments chapter 7 have to be subsumed.

For the reference parametrizations ES and ID conclusions are drawn, that go in common with the previous work in certain parts. For the ES parametrization the inconsistencies and instabilities, that have already discussed in literature, apply. Except for the discussion made by Bailey et al. in [Bai+06] the research community's opinion is, that the ES parametrization is consistent for nearby landmarks. Nevertheless, this thesis' analysis showed erroneous stochastic models in the presence of highly uncertain pose orientations. This conclusion is similar to the observations made by Bailey et al. Moreover, due to the inconsistent and non informative covariances in ES, an estimation of the IMU to camera calibration is impossible. Even though the ES estimator provided a precise plane navigation for the Rawseeds dataset, it's estimation quality in 6 DOF is degenerated compared to ID and PD.

As was discussed in literature the ID representation outperforms the parametrization of landmarks in ES. This holds especially for the uncertainty representation for the camera to landmark distance. Beside the significant improvement to the filter covariance's informativity it is capable of stably performing the IMU to camera auto calibration in parallel to SLAM. Using linear estimators with ID landmarks allows for a stable estimation of the system pose and the environment map even for long term navigation. Nevertheless, the neglected non linearities in IDs initialization of the landmarks view ray cause significant inconsistencies for the system's pose. Even though the pose's informativity is significantly improved in comparison to ES, the inlier to outlier ratio still deviates from the true ratio by $15 - 30\%$ for long term navigation. To the author's knowledge this effect and the respective non linearity have not yet been discussed in literature.

In contrast to ES and ID the PD representation was able to estimate a fully consistent navigation state and IMU to camera calibration even in long term navigation. The only constraint to be made is the validity of the prediction model. As for ES and ID the informativity for the state covariance is disturbed by the neglected inertial sub models for the second offline test series. Nevertheless, it can be assumed, that this effect is caused by the inconsistent prediction model solely. Due to the simple landmark

initialization model environment maps of millions of landmarks can be created for trajectories of not more than 30 minutes. The availability of a simple, relative adjustment model allows for a reconstruction of the traveled trajectory and the environment map with high precision. In especially the implicit improvement of large environment maps is a major advantage compared to other adjustment techniques like bundle adjustment. Nevertheless, the process of state relocalization increases the effort of filter management. Moreover, landmarks close to the camera plane cause significant linearization errors and have to be removed from the filter state. To subsume, the newly proposed PD landmark representation outperforms even the state of the art methods as the ID representation in consistency for long term estimation.

The main conclusion of this thesis is, that a consistent and precise estimation of long term navigation is possible by using an IMU aided stereo SLAM system. Moreover, an inclusion of the IMU to camera calibration delivers satisfactory results. The ID and PD estimators leave a pre-calibration non necessary providing that the calibration is known with a precision of up to several degrees.

### Future Work

Since the landmark parametrization PD has been designed from scratch, it is worth considering improvements for the respective models. The most obvious task is, to solve the problem of relocalization of out of sight landmarks. As discussed, significant linearization errors are incorporated to the relocalization of landmarks, that are close to the camera plane. Thus, methods have to be deduced, that allow for the consistent transfer of such landmarks. By this, the administrative costs for keeping inactive landmarks in the filter state can be reduced.

Another important task of further research is the augmentation of the movement prediction by advanced inertial navigation modeling. The simplification of these models has been found, to be a major cause of inconsistency in IMU aided SLAM. Thus, taking the earths turn rate and other complex inertial models into account, is expected to improve the long term stability of the system state.

A further subject to future work is the implementation of an online adjustment by loop closing. For this purpose the proposed adjustment

model has to be embedded in an loop close detection module. Thanks to the incremental representation of system poses such adjustments would impact all poses succeeding the closed sub track. Moreover, introducing hierarchical methods to this estimation process could speed up the adjustment process.

To reduce the negative effects of non overlapping views in stereo vision, updates for single image features can be embedded to the filter framework. That is, after a PD landmark has been created, observations by only a single camera can be applied in the KF update. Moreover, using points at infinity provides the opportunity, to improve the orientation estimation. This model improvement has been previously proposed by Montiel and Davison in [MD06] for ID parametrizations. Such features are characterized by a disparity and inverse depth equal to 0. By this, a PD landmark at infinity would reduce to a representation by a single 2D position (using $x^{\mathrm{PD}}$ only). It has to be evaluated, whether this state reduction has an impact to the systems consistency.

Appendix A

# Derivations

## A.1 Observations using TFT

This section describes the derivation of the basic PD observation model (see section 4.2.6, page 83, definitions (4.2.41) and (4.2.44)). For this purpose the tensor field notation and Einstein's Sum Convention ([BS96]) is used. The notation and definitions for the trifocal tensor have been taken from [HZ03]. Let

$$\begin{aligned} P &= \left( \; 1^3 \;\middle|\; 0_3 \; \right) \\ P' &= \left( \; A \;\middle|\; a_4 \; \right) \quad \text{with} \quad A = \left( \; a_1 \;\middle|\; a_2 \;\middle|\; a_3 \; \right) \\ P'' &= \left( \; B \;\middle|\; b_4 \; \right) \quad \text{with} \quad B = \left( \; b_1 \;\middle|\; b_2 \;\middle|\; b_3 \; \right) \end{aligned}$$

$$\mathbf{P} = [P] \quad \mathbf{P}' = \left[P'\right] \quad \mathbf{P}'' = \left[P''\right]$$

the projection matrices for three cameras, where the first $\mathbf{P}$ is assumed to be the reference coordinate frame (i. e., frame $n$ in section 4.2.6). The TFT $\mathcal{T}$ can be used, to map corresponding points from $\mathbf{P}$ and $\mathbf{P}'$ respectively to $\mathbf{P}''$. It can be determined by:

$$\mathcal{T}_{\mathtt{i}}^{\mathtt{jk}} = a_{\mathtt{i}}^{\mathtt{j}} b_4^{\mathtt{k}} - a_4^{\mathtt{j}} b_{\mathtt{i}}^{\mathtt{k}} \quad \forall \mathtt{i}, \mathtt{j}, \mathtt{k} \in \{1, 2, 3\}$$

Because no rotation between master and slave cameras in a stereo rig is assumed, it holds $A = 1^3$. By this, it follows:

$$\begin{aligned} \mathtt{i} = \mathtt{j} \quad &\Rightarrow \quad \mathcal{T}_{\mathtt{i}}^{\mathtt{jk}} = b_4^{\mathtt{k}} - a_4^{\mathtt{j}} b_{\mathtt{i}}^{\mathtt{k}} \\ \mathtt{i} \neq \mathtt{j} \quad &\Rightarrow \quad \mathcal{T}_{\mathtt{i}}^{\mathtt{jk}} = -a_4^{\mathtt{j}} b_{\mathtt{i}}^{\mathtt{k}} \end{aligned}$$

Moreover, the translation between master and slave camera is restricted to the $x$-axis, that is $a_4 = \left(a_4^1, 0, 0\right)^T$ with $a_4^1 \neq 0$. Thus:

$$\begin{aligned}
\mathtt{i} = \mathtt{j} \wedge \mathtt{j} = 1 \quad &\Rightarrow \quad \mathcal{T}_{\mathtt{i}}^{\mathtt{jk}} = b_4^{\mathtt{k}} - a_4^{\mathtt{j}} b_{\mathtt{i}}^{\mathtt{k}} \\
\mathtt{i} = \mathtt{j} \wedge \mathtt{j} \neq 1 \quad &\Rightarrow \quad \mathcal{T}_{\mathtt{i}}^{\mathtt{jk}} = b_4^{\mathtt{k}} \\
\mathtt{i} \neq \mathtt{j} \wedge \mathtt{j} = 1 \quad &\Rightarrow \quad \mathcal{T}_{\mathtt{i}}^{\mathtt{jk}} = -a_4^{\mathtt{j}} b_{\mathtt{i}}^{\mathtt{k}} \\
\mathtt{i} \neq \mathtt{j} \wedge \mathtt{j} \neq 1 \quad &\Rightarrow \quad \mathcal{T}_{\mathtt{i}}^{\mathtt{jk}} = 0
\end{aligned}$$

Let $[x], [x'] \in \mathbb{P}^2$ the images of a point $\mathbf{X} \in \mathbb{P}^3$ mapped by $\mathbf{P}$ and $\mathbf{P}'$ respectively. $\mathbf{X}$ is assumed not to lie on the line joining both camera centers. This is feasible since no rotation between $\mathbf{P}$ and $\mathbf{P}'$ is assumed, thus, cameras will not observe such points. From this it follows that $x^3 \neq 0 \neq x'^3$. Let:

$$\begin{aligned}
d &= \frac{x^1}{x^3} - \frac{x'^1}{x'^3} \\
l' &= \left(x'^3, 0, -x'^1\right) \quad \text{that is} \quad 0 = l'_{\mathtt{j}} x'^{\mathtt{j}}
\end{aligned}$$

By this $[l']$ is perpendicular to the image rows (epipolar line of $[x]$ in $\mathbf{P}'$) and ensures the applicability of equation (2.3.26) on page 26. Following [HZ03] chapter 15.3.2, $\mathcal{T}$ maps $[x]$ and $[l']$ to the point $[x'']$ for $\mathbf{P}''$ corresponding to $[x]$ and $[x']$:

$$\begin{aligned}
x''^k &= x^{\mathtt{i}} l'_{\mathtt{j}} \mathcal{T}_{\mathtt{i}}^{\mathtt{jk}} \\
&= x^{\mathtt{i}} \left(x'^3, 0, -x'^1\right)_{\mathtt{j}} \mathcal{T}_{\mathtt{i}}^{\mathtt{jk}} \\
&= x^{\mathtt{i}} x'^3 \mathcal{T}_{\mathtt{i}}^{1\mathtt{k}} - x^{\mathtt{i}} x'^1 \mathcal{T}_{\mathtt{i}}^{3\mathtt{k}} \\
&= x^{\mathtt{i}} x'^3 \mathcal{T}_{\mathtt{i}}^{1\mathtt{k}} - x^{\mathtt{i}} x'^3 \left(\frac{x^1}{x^3} - d\right) \mathcal{T}_{\mathtt{i}}^{3\mathtt{k}} \\
&= x'^3 \left(x^{\mathtt{i}} \mathcal{T}_{\mathtt{i}}^{1\mathtt{k}} + x^{\mathtt{i}} d \mathcal{T}_{\mathtt{i}}^{3\mathtt{k}} - x^{\mathtt{i}} \frac{x^1}{x^3} \mathcal{T}_{\mathtt{i}}^{3\mathtt{k}}\right) \\
&= x'^3 \left(x^1 \mathcal{T}_1^{1\mathtt{k}} + x^2 \mathcal{T}_2^{1\mathtt{k}} + x^3 \mathcal{T}_3^{1\mathtt{k}} + x^3 d \mathcal{T}_3^{3\mathtt{k}} - x^3 \frac{x^1}{x^3} \mathcal{T}_3^{3\mathtt{k}}\right) \\
&= x'^3 \left(x^1 \left(b_4^{\mathtt{k}} - a_4^1 b_1^{\mathtt{k}}\right) - x^2 a_4^1 b_2^{\mathtt{k}} - x^3 a_4^1 b_3^{\mathtt{k}} + x^3 d b_4^{\mathtt{k}} - x^3 \frac{x^1}{x^3} b_4^{\mathtt{k}}\right)
\end{aligned}$$

$$
\begin{aligned}
&= x'^3\left(x^1b_4^{\mathtt{k}} - x^1a_4^1b_1^{\mathtt{k}} - x^2a_4^1b_2^{\mathtt{k}} - x^3a_4^1b_3^{\mathtt{k}} + x^3db_4^{\mathtt{k}} - x_1b_4^{\mathtt{k}}\right)\\
&= x'^3\left(-x^1a_4^1b_1^{\mathtt{k}} - x^2a_4^1b_2^{\mathtt{k}} - x^3a_4^1b_3^{\mathtt{k}} + x^3db_4^{\mathtt{k}}\right)\\
&= -x'^3\left(x^1a_4^1b_1^{\mathtt{k}} + x^2a_4^1b_2^{\mathtt{k}} + x^3a_4^1b_3^{\mathtt{k}} - x^3db_4^{\mathtt{k}}\right)\\
&= -x'^3a_4^1\left(x^1b_1^{\mathtt{k}} + x^2b_2^{\mathtt{k}} + x^3b_3^{\mathtt{k}} - x^3\frac{d}{a_4^1}b_4^{\mathtt{k}}\right)
\end{aligned}
$$

And finally:

$$
\begin{aligned}
x'' &= -x'^3a_4^1\begin{pmatrix} x^1b_1^1 + x^2b_2^1 + x^3b_3^1 - x^3\frac{d}{a_4^1}b_4^1\\ x^1b_1^2 + x^2b_2^2 + x^3b_3^2 - x^3\frac{d}{a_4^1}b_4^2\\ x^1b_1^3 + x^2b_2^3 + x^3b_3^3 - x^3\frac{d}{a_4^1}b_4^3\end{pmatrix}\\
&= -x'^3a_4^1\left(x^1b_1 + x^2b_2 + x^3b_3 - \frac{d}{a_4^1}x^3b_4\right)\\
&= -x'^3a_4^1P''\begin{pmatrix} x^1\\ x^2\\ x^3\\ -\frac{d}{a_4^1}x^3\end{pmatrix}\\
&= -x'^3a_4^1B\begin{pmatrix} x^1\\ x^2\\ x^3\end{pmatrix} + dx'^3x^3b_4 \qquad \text{(A.1.1)}
\end{aligned}
$$

Further let $P''' = \left(\begin{array}{c|c} B & b_4 + a_4 \end{array}\right)$ and $[x''']$ the respective correspondence to $[x]$ and $[x']$ respectively. It follows:

$$
\begin{aligned}
x''' &= -x'^3a_4^1P'''\begin{pmatrix} x^1\\ x^2\\ x^3\\ -\frac{d}{a_4^1}x^3\end{pmatrix}\\
&= -x'^3a_4^1\left[B|b_4 + a_4\right]\begin{pmatrix} x^1\\ x^2\\ x^3\\ -\frac{d}{a_4^1}x^3\end{pmatrix}
\end{aligned}
$$

$$
\begin{aligned}
&= -x'^3 a_4^1 B \begin{pmatrix} x^1 \\ x^2 \\ x^3 \end{pmatrix} + dx'^3 x^3 b_4 + dx'^3 x^3 a_4 \\
&= x'' + \begin{pmatrix} da_4^1 x'^3 x^3 \\ 0 \\ 0 \end{pmatrix}
\end{aligned} \tag{A.1.2}
$$

## A.2 Full Landmark Models

In this section the full landmark models are defined. That is, the coordinate transformation $\Gamma_c^i = \langle \Phi_c^i, t_{ci}^c \rangle$ for the IMU-camera calibration is embedded in the models in sections 4.2.4 (pages 77ff) and 4.2.6 (pages 80ff).

### A.2.1 Initialization

#### ES

$$
\begin{aligned}
\mathcal{X}^{\text{ES}} &= t_{eX}^e = \Gamma_i^e \circ \Gamma_i^{cT}\left(t_{cX}^c\right) \\
&\overset{(2.2.5)}{=} \Gamma_i^e \left( \Phi_i^{cT} \cdot t_{cX}^c - \Phi_i^{cT} \cdot t_{ci}^c \right) \\
&\overset{(2.2.5)}{=} \Phi_i^e \left( \Phi_i^{cT} \cdot t_{cX}^c - \Phi_i^{cT} \cdot t_{ci}^c \right) - \Phi_i^e \cdot t_{ie}^i \\
&\overset{(2.2.5)}{=} \Phi_i^e \cdot \Phi_i^{cT} \cdot (t_{cX}^c - t_{ci}^c) + t_{ei}^e \\
&= \Phi_i^e \cdot \Phi_i^{cT} \cdot \left( \frac{b}{x_{(1)} - y_{(1)}} \cdot \begin{pmatrix} x_{(1)} \\ \frac{1}{2}\left(x_{(2)} + y_{(2)}\right) \\ 1 \end{pmatrix} - t_{ci}^c \right) + t_{ei}^e
\end{aligned} \tag{A.2.1}
$$

**ID** Following 4.2.1 on pages 69ff the ID landmark is $\mathcal{X}^{\text{ID}} = (\, C \;\; \phi \;\; \psi \;\; \rho \,)$:

$$
\mathcal{C} = t_{ec}^e = t_{ei}^e + t_{ic}^e = t_{ei}^e - \Phi_i^e \cdot \Phi_i^{cT} \cdot t_{ci}^c \tag{A.2.2}
$$

$$
\rho = |t_{cX}^c|_2^{-1} = \left| \frac{b}{x_{(1)} - y_{(1)}} \cdot \begin{pmatrix} x_{(1)} \\ \frac{1}{2}\left(x_{(2)} + y_{(2)}\right) \\ 1 \end{pmatrix} \right|_2^{-1} \tag{A.2.3}
$$

$$= \frac{x_{(1)} - y_{(1)}}{b} \cdot \left( {x_{(1)}}^2 + \frac{1}{4} \cdot \left( x_{(2)} + y_{(2)} \right)^2 + 1 \right)^{-\frac{1}{2}} \tag{A.2.4}$$

Using the definition

$$z = \Phi_i^e \Phi_i^{cT} \begin{pmatrix} x_{(1)} \\ \frac{1}{2}\left( x_{(2)} + y_{(2)} \right) \\ 1 \end{pmatrix} \tag{A.2.5}$$

the spherical coordinates $\phi$ and $\psi$ for the global view rays representation are:

$$\phi = \arctan_2\left( z_{(1)}, z_{(3)} \right) \tag{A.2.6}$$

$$\psi = \arctan_2\left( -z_{(2)}, \sqrt{z_{(1)} + z_{(3)}} \right) \tag{A.2.7}$$

**PD** is independent of the camera pose. Thus, the initialization of a PD landmark is not effected by the IMU-camera calibration and equation (4.2.18) on page 79 stays unchanged.

## A.2.2 Observation

### ES

$$\begin{aligned} \mathbf{x}^{\text{ES}} &\overset{(4.2.24)}{=} [R_e^c \cdot (\mathcal{X}^{\text{ES}} - t_{ec}^e)] = \left[ \Phi_i^c \cdot \Phi_i^{eT} \cdot (\mathcal{X}^{\text{ES}} - t_{ei}^e - t_{ic}^e) \right] \\ &= \left[ \Phi_i^c \cdot \Phi_i^{eT} \cdot (\mathcal{X}^{\text{ES}} - t_{ei}^e + t_{ci}^e) \right] \\ &= \left[ \Phi_i^c \cdot \Phi_i^{eT} \cdot (\mathcal{X}^{\text{ES}} - t_{ei}^e) + t_{ci}^c \right] \end{aligned} \tag{A.2.8}$$

$$\begin{aligned} \mathbf{y}^{\text{ES}} &\overset{(4.2.25)}{=} [R_e^c \cdot (\mathcal{X}^{\text{ES}} - t_{ec}^e) - t_{cc'}^c] \\ &= \left[ \Phi_i^c \cdot \Phi_e^{iT} \cdot (\mathcal{X}^{\text{ES}} - t_{ei}^e) + t_{ci}^c - t_{cc'}^c \right] \end{aligned} \tag{A.2.9}$$

### ID

$$\begin{aligned} \mathbf{x}^{\text{ID}} &\overset{(4.2.29)}{=} [R_e^c \cdot (ray\,(\phi, \psi) + \rho \cdot (\mathcal{C} - t_{ec}^e))] \\ &= \left[ \Phi_i^c \cdot \Phi_i^{eT} \cdot (ray\,(\phi, \psi) + \rho \cdot (\mathcal{C} - t_{ei}^e - t_{ic}^e)) \right] \end{aligned}$$

$$
\begin{aligned}
&= \left[\Phi_i^c \cdot \Phi_i^{eT} \cdot (ray\,(\phi,\psi) + \rho \cdot (\mathcal{C} - t_{ei}^e)) - \rho \cdot \Phi_i^c \cdot \Phi_i^{eT} \cdot t_{ic}^e\right] \\
&= \left[\Phi_i^c \cdot \Phi_i^{eT} \cdot (ray\,(\phi,\psi) + \rho \cdot (\mathcal{C} - t_{ei}^e)) - \rho \cdot t_{ic}^c\right] \\
&= \left[\Phi_i^c \cdot \Phi_i^{eT} \cdot (ray\,(\phi,\psi) + \rho \cdot (\mathcal{C} - t_{ei}^e)) + \rho \cdot t_{ci}^c\right] \qquad \text{(A.2.10)}
\end{aligned}
$$

$$
\begin{aligned}
\mathbf{y}^{\text{ID}} \overset{(4.2.30)}{=}\ & [R_e^c \cdot (ray\,(\phi,\psi) + \rho \cdot (\mathcal{C} - t_{ec}^e)) - \rho \cdot t_{cc'}^c] \\
=\ & \left[\Phi_i^c \cdot \Phi_i^{eT} \cdot (ray\,(\phi,\psi) + \rho \cdot (\mathcal{C} - t_{ei}^e)) + \rho \cdot (t_{ci}^c - t_{cc'}^c)\right] \qquad \text{(A.2.11)}
\end{aligned}
$$

**PD** To determine the full observation model for PD parametrization, the assumption $\Gamma_{c_o}^n = \langle 1^3, 0_3 \rangle = \Gamma_c^i$, used to simplify equations (4.2.32) and (4.2.34), cannot be applied. Although $\Gamma_{c_o}^n = \Gamma_c^i$ still holds, $\langle \Phi_i^c, t_{ci}^c \rangle$ has to be added in (4.2.37) and (4.2.38) (see page 84).

To do so, the coordinate transformations $\Gamma_{c_o}^c$ and $\Gamma_{c_o}^{c'}$ can be simplified as follows:

$$
\begin{aligned}
R_{c_0}^c &= R_i^c \cdot R_n^i \cdot R_{c_0}^n = R_i^c \cdot R_i^{nT} \cdot R_i^{cT} \qquad \text{(A.2.12)} \\
t_{c_o c}^{c_o} &= t_{c_o n}^{c_o} + t_{ni}^{c_o} + t_{ic}^{c_o} \\
&= \left(R_c^{c_o} \cdot R_{c_o}^c\right) \cdot (t_{ci}^c + R_n^{c_o} \cdot t_{ni}^n + R_c^{c_o} \cdot t_{ic}^c) \\
&= R_c^{c_o} \cdot \left(R_{c_o}^c \cdot t_{ci}^c + R_n^c \cdot t_{ni}^n + R_c^c \cdot t_{ic}^c\right) \\
&= R_c^{c_o} \cdot \left(R_{c_o}^c \cdot t_{ci}^c + R_i^c \cdot R_n^i \cdot t_{ni}^n - t_{ci}^c\right)
\end{aligned}
$$

Moreover, the stereo constraint $R_c^{c'} = R_{c_o}^{c'} = 1^3$ implies $R_{c_0}^{c'} = R_{c_0}^c$ and $t_{c_o c'}^{c_o} = t_{c_o c}^{c_o} + R_c^{c_o} \cdot t_{cc'}^c$. By this, the new definitions for $P''$ and $P'''$ are:

$$
\begin{aligned}
\Gamma_{c_o}^c : P'' &= \left[R_{c_0}^c \middle| - R_{c_0}^c\, t_{c_o c}^{c_o}\right] \\
&= \left[R_i^c \cdot R_n^i \cdot R_c^i \middle| - R_{c_0}^c \cdot R_c^{c_o} \cdot \left(R_{c_o}^c \cdot t_{ci}^c + R_i^c \cdot R_n^i \cdot t_{ni}^n - t_{ci}^c\right)\right] \\
&= \left[R_i^c \cdot R_n^i \cdot R_c^i \middle| - R_{c_o}^c \cdot t_{ci}^c - R_i^c \cdot R_n^i \cdot t_{ni}^n + t_{ci}^c\right] \qquad \text{(A.2.13)} \\
\Gamma_{c_o}^{c'} : P'' &= \left[R_i^c \cdot R_n^i \cdot R_c^i \middle| - R_{c_o}^c \cdot t_{ci}^c - R_i^c \cdot R_n^i \cdot t_{ni}^n + t_{ci}^c - R_{c_0}^c \cdot R_c^{c_o} \cdot t_{cc'}^c\right] \\
&= \left[R_i^c \cdot R_n^i \cdot R_c^i \middle| - R_{c_o}^c \cdot t_{ci}^c - R_i^c \cdot R_n^i \cdot t_{ni}^n + t_{ci}^c - t_{cc'}^c\right] \qquad \text{(A.2.14)}
\end{aligned}
$$

As in section 4.2.6 (page 84) $P''$ and $P'''$ are substituted in equation (A.1.1) on page 217. Using $\mathcal{X}^{\mathrm{PD}} = \begin{pmatrix} x^{\mathrm{PD}} & d \end{pmatrix}$ to define

$$x = \begin{pmatrix} x^{\mathrm{PD}} \\ 1 \end{pmatrix} \quad \text{and} \quad x' = \begin{pmatrix} x^{\mathrm{PD}}(1) - d \\ x^{\mathrm{PD}}(2) \\ 1 \end{pmatrix} \tag{A.2.15}$$

the augmented observation model for PD landmarks results in:

$$\begin{aligned}
\mathbf{x}^{\mathrm{PD}} &\overset{(A.1.1)}{=} \left[-x'(3)\cdot b\cdot R^c_{c_o}\cdot x - d\cdot x(3)\cdot x'(3)\cdot\left(-R^c_{c_o}\cdot t^{c_o}_{c_o c}\right)\right] \\
&= \left[d\cdot R^c_{c_o}\cdot R^{c_o}_c\cdot\left(R^c_{c_o}\cdot t^c_{ci} + R^c_i\cdot R^i_n\cdot t^n_{ni} - t^c_{ci}\right) - b\cdot R^c_{c_o}\cdot x\right] \\
&= \left[d\cdot R^c_{c_o}\cdot t^c_{ci} + d\cdot R^c_{c_o}\cdot R^{c_o}_c\cdot R^c_i\cdot R^i_n\cdot t^n_{ni} - d\cdot t^c_{ci} - b\cdot R^c_{c_o}\cdot x\right] \\
&\overset{(A.2.12)}{=} \left[d\cdot R^c_i\cdot R^i_n\cdot R^i_c\cdot t^c_{ci} + dR^c_i\cdot R^i_n\cdot t^n_{ni} - d\cdot t^c_{ci} - b\cdot R^c_i\cdot R^i_n\cdot R^i_c\cdot x\right] \\
&= \left[R^c_i\cdot R^i_n\cdot\left(d\cdot R^i_c\cdot t^c_{ci} + d\cdot t^n_{ni} - b\cdot R^i_c\cdot x\right) - d\cdot t^c_{ci}\right] \\
&= \left[R^c_i\cdot R^i_n\cdot\left(R^i_c\cdot(d\cdot t^c_{ci} - b\cdot x) + d\cdot t^n_{ni}\right) - d\cdot t^c_{ci}\right] \\
&\overset{(*)}{=} \left[R^c_i\cdot R^{nT}_i\cdot\left(\Phi^{cT}_i\cdot(b\cdot x - d\cdot t^c_{ci}) - d\cdot t^n_{ni}\right) + d\cdot t^c_{ci}\right] \\
&= \left[\Phi^c_i\cdot\Phi^{nT}_i\cdot\left(\Phi^{cT}_i\cdot(b\cdot x - d\cdot t^c_{ci}) - d\cdot t^n_{ni}\right) + d\cdot t^c_{ci}\right]
\end{aligned} \tag{A.2.16}$$

$$\begin{aligned}
\mathbf{y}^{\mathrm{PD}} &\overset{(A.1.2)}{=} \left[-x'(3)\cdot b\cdot R^c_{c_o}\cdot x - d\cdot x(3)\cdot x'(3)\cdot\left(-R^c_{c_o}\cdot t^{c_o}_{c_o c}\right) + \begin{pmatrix} d\cdot b\cdot x'^3\cdot x^3 \\ 0 \\ 0 \end{pmatrix}\right] \\
&= \left[R^c_i\cdot R^i_n\cdot\left(R^i_c\cdot(d\cdot t^c_{ci} - b\cdot x) + d\cdot t^n_{ni}\right) - d\cdot\left(t^c_{ci} - \begin{pmatrix} b \\ 0 \\ 0 \end{pmatrix}\right)\right] \\
&= \left[R^c_i\cdot R^i_n\cdot\left(R^i_c\cdot(d\cdot t^c_{ci} - b\cdot x) + d\cdot t^n_{ni}\right) - d\cdot(t^c_{ci} - t^c_{cc'})\right] \\
&\overset{(*)}{=} \left[R^c_i\cdot R^{nT}_i\cdot\left(R^{cT}_i\cdot(b\cdot x - d\cdot t^c_{ci}) - d\cdot t^n_{ni}\right) + d\cdot(t^c_{ci} - t^c_{cc'})\right] \\
&= \left[\Phi^c_i\cdot\Phi^{nT}_i\cdot\left(\Phi^{cT}_i\cdot(b\cdot x - d\cdot t^c_{ci}) - d\cdot t^n_{ni}\right) + d\cdot(t^c_{ci} - t^c_{cc'})\right]
\end{aligned} \tag{A.2.17}$$

The equalities marked with $(*)$ were applied using the scale invariance, to visualize the similarities of the observation models ES, ID and PD. Note, that as for the models not including the camera-IMU calibration $\Gamma^c_i = \langle \Phi^c_i, t^c_{ci} \rangle$ the equations for PD and ID mainly differ in the view rays representation.

That is the linear term $b \cdot x$ versus the non linear term $ray\,(\phi, \psi)$. Another noticeable fact is, that $\Gamma_i^c$ is used twice in the PD observation (backward and forward transformation). In contrast to this, the ES and ID parametrization use $\Gamma_i^c$ backward for initialization and forward for observation of landmarks.

## A.3 Jacobians

This section introduces the Jacobian matrices for the landmarks initialization (appendix A.2.1) and observation (appendix A.2.2) models.

### A.3.1 Initialization Models

As for the derivation of the observation models keep in mind that $t_{cc'}^c = \begin{pmatrix} b & 0 & 0 \end{pmatrix}^T$, due to the stereo constraints.

#### ES initialization

To determine the derivatives for the initialization of ES landmarks (see A.2.1, page 218) let:

$$\begin{aligned} g\,(x, y, t_{ci}^c) &:= \frac{b}{x(1) - y(1)} \cdot \begin{pmatrix} x(1) \\ \frac{1}{2} \cdot (x(2) + y(2)) \\ 1 \end{pmatrix} - t_{ci}^c \\ d &:= x(1) - y(1) \quad \text{and} \quad \mathtt{x} := \frac{b}{d} \cdot \begin{pmatrix} x(1) \\ \frac{1}{2} \cdot (x(2) + y(2)) \\ 1 \end{pmatrix} \end{aligned}$$

Where $x$, $y$ are the observed stereo correspondences used for initialization. Using this the derivatives for $\mathcal{X}^{\text{ES}}$ by the global pose $\langle \Phi_i^e, t_{ei}^e \rangle$, the camera-IMU calibration $\langle \Phi_c^i, t_{ci}^c \rangle$ and the observed stereo correspondences $x$ and $y$ are:

$$\frac{\partial \mathcal{X}^{\text{ES}}}{\partial t_{ei}^e} = 1^3 \tag{A.3.1}$$

$$\frac{\partial \mathcal{X}^{\text{ES}}}{\partial \Phi_i^e} = \Phi_i^e \cdot \left[ \frac{\partial R(\phi)}{\partial \phi(1)}|_0 \;\middle|\; \frac{\partial R(\phi)}{\partial \phi(2)}|_0 \;\middle|\; \frac{\partial R(\phi)}{\partial \phi(3)}|_0 \right] \cdot \Phi_i^{cT} \cdot g\left(x, y, t_{ci}^c\right) \tag{A.3.2}$$

$$\frac{\partial \mathcal{X}^{\text{ES}}}{\partial t_{ci}^c} = \Phi_i^e \cdot \Phi_i^{cT} \cdot \frac{\partial g}{\partial t_{ci}^c} = -\Phi_i^e \cdot \Phi_i^{cT} \tag{A.3.3}$$

$$\frac{\partial \mathcal{X}^{\text{ES}}}{\partial \Phi_i^c} = \Phi_i^e \cdot \left[ \frac{\partial R(\phi)}{\partial \phi(1)}|_0 \;\middle|\; \frac{\partial R(\phi)}{\partial \phi(2)}|_0 \;\middle|\; \frac{\partial R(\phi)}{\partial \phi(3)}|_0 \right]^T \cdot \Phi_i^{cT} \cdot g\left(x, y, t_{ci}^c\right) \tag{A.3.4}$$

$$\begin{aligned} \frac{\partial \mathcal{X}^{\text{ES}}}{\partial x} &= \Phi_i^e \cdot \Phi_i^{cT} \cdot \left. \frac{\partial g\left(\cdot, y, t_{ci}^c\right)}{\partial} \right|_x \\ &= \Phi_i^e \cdot \Phi_i^{cT} \cdot \left( -\frac{1}{x(1) - y(1)} \cdot \mathtt{x} + \frac{b}{d} \cdot e_1^3 \;\middle|\; \frac{b}{x(1) - y(1)} \cdot \begin{pmatrix} 0 \\ \frac{1}{2} \\ 0 \end{pmatrix} \right) \\ &= \frac{b}{d} \cdot \Phi_i^e \cdot \Phi_i^{cT} \cdot \left( e_1^3 - \frac{1}{b} \cdot \mathtt{x} \;\middle|\; \tfrac{1}{2} \cdot e_2^3 \right) \end{aligned} \tag{A.3.5}$$

$$\begin{aligned} \frac{\partial \mathcal{X}^{\text{ES}}}{\partial y} &= \Phi_i^e \cdot \Phi_i^{cT} \cdot \left. \frac{\partial g\left(x, \cdot, t_{ci}^c\right)}{\partial} \right|_y \\ &= \Phi_i^e \cdot \Phi_i^{cT} \cdot \left( \frac{1}{x(1) - y(1)} \cdot \mathtt{x} \;\middle|\; \frac{b}{x(1) - y(1)} \cdot \begin{pmatrix} 0 \\ \frac{1}{2} \\ 0 \end{pmatrix} \right) \\ &= \frac{b}{d} \cdot \Phi_i^e \cdot \Phi_i^{cT} \cdot \left( \frac{1}{b} \cdot \mathtt{x} \;\middle|\; \tfrac{1}{2} \cdot e_2^3 \right) \end{aligned} \tag{A.3.6}$$

#### ID initialization

Determining the Jacobian matrix for ID initialization is done in three steps. The first is the derivative for the landmark position $\mathcal{C}$, the second is for the view rays angles $\phi, \psi$ and the last for the inverse depth $\rho$.

For $\mathcal{C}$ (see (A.2.2), page 218) the derivative can be computed straight forward:

$$\left. \frac{\partial \mathcal{C}}{\partial t_{ei}^e} \right| = 1^3 \tag{A.3.7}$$

$$\left.\frac{\partial \mathcal{C}}{\partial \Phi_i^e}\right| = -\Phi_i^e \cdot \left[ \frac{\partial R(\phi)}{\partial \phi(1)}|_0 \;\middle|\; \frac{\partial R(\phi)}{\partial \phi(2)}|_0 \;\middle|\; \frac{\partial R(\phi)}{\partial \phi(3)}|_0 \right] \cdot \Phi_i^{cT} \cdot t_{ci}^c \tag{A.3.8}$$

$$\left.\frac{\partial \mathcal{C}}{\partial t_{ci}^c}\right| = -\Phi_i^e \cdot \Phi_i^{cT} \tag{A.3.9}$$

$$\left.\frac{\partial \mathcal{C}}{\partial \Phi_c^i}\right| = -\Phi_i^e \cdot \left[ \frac{\partial R(\phi)}{\partial \phi(1)}|_0 \;\middle|\; \frac{\partial R(\phi)}{\partial \phi(2)}|_0 \;\middle|\; \frac{\partial R(\phi)}{\partial \phi(3)}|_0 \right]^T \cdot \Phi_i^{cT} \cdot t_{ci}^c \tag{A.3.10}$$

To ease the readability of the following derivations let

$$f(g) = \begin{cases} \arctan\left(\frac{g(1)}{g(2)}\right) & : \quad g(2) > 0 \\ \operatorname{sign}(g(1)) \cdot \frac{\pi}{2} & : \quad g(2) = 0 \\ \arctan\left(\frac{g(1)}{g(2)}\right) + \operatorname{sign}(g(1)) \cdot \pi & : \quad g(2) < 0 \end{cases}$$

$$= \arctan_2(g(1), g(2)) \tag{A.3.11}$$

$$g_1(h) = \begin{pmatrix} h(1) \\ h(3) \end{pmatrix}$$

$$g_2(h) = \begin{pmatrix} -h(2) \\ \sqrt{h(1)^2 + h(3)^2} \end{pmatrix}$$

$$h(\Phi_i^e, \Phi_i^c, \mathtt{x}) = \Phi_i^e \cdot \Phi_i^{cT} \cdot \mathtt{x}$$

$$\mathtt{x}(x, y) = \frac{b}{x(1) - y(1)} \cdot \begin{pmatrix} x(1) \\ \frac{1}{2} \cdot (x(2) + y(2)) \\ 1 \end{pmatrix}$$

Using this the view ray $\phi, \psi$ is computed by:

$$\phi = (f \circ g_1)(h(\Phi_i^e, \Phi_i^c, \mathtt{x}(x,y))) \tag{A.3.12}$$

$$\psi = (f \circ g_2)(h(\Phi_i^e, \Phi_i^c, \mathtt{x}(x,y))) \tag{A.3.13}$$

Note that due to the scale invariance of $f = \arctan_2$ the factor $b \cdot (x(1) - y(1))^{-1}$ in $\mathtt{x}$ can be omitted in equation (A.3.12) and (A.3.13) respectively. In the following, $g_1$, $g_2$, $h$ and $\mathtt{x}$ are used as abbreviations for $g_1(h(\Phi_i^e, \Phi_i^c, \mathtt{x}(x,y)))$

etc. By this the respective Jacobians are given by:

$$\left.\frac{\partial\left[\psi|\phi\right]}{\partial\Phi_i^e}\right|_{\Phi_i^e,\Phi_i^c,x,y} = \left.\frac{\partial f}{\partial}\right|_{[g_1|g_2]} \cdot \left.\frac{\partial\left[g_1|g_2\right]}{\partial}\right|_h \cdot \left.\frac{\partial h}{\partial\Phi_i^e}\right|_{\Phi_i^e,\Phi_i^c,\mathtt{x}} \tag{A.3.14}$$

$$\left.\frac{\partial\left[\psi|\phi\right]}{\partial\Phi_i^c}\right|_{\Phi_i^e,\Phi_i^c,x,y} = \left.\frac{\partial f}{\partial}\right|_{[g_1|g_2]} \cdot \left.\frac{\partial\left[g_1|g_2\right]}{\partial}\right|_h \cdot \left.\frac{\partial h}{\partial\Phi_i^c}\right|_{\Phi_i^e,\Phi_i^c,\mathtt{x}} \tag{A.3.15}$$

$$\left.\frac{\partial\left[\psi|\phi\right]}{\partial x,y}\right|_{\Phi_i^e,\Phi_i^c,x,y} = \left.\frac{\partial f}{\partial}\right|_{[g_1|g_2]} \cdot \left.\frac{\partial\left[g_1|g_2\right]}{\partial}\right|_h \cdot \left.\frac{\partial h}{\partial\mathtt{x}}\right|_{\Phi_i^e,\Phi_i^c,\mathtt{x}} \cdot \left.\frac{\partial\mathtt{x}}{\partial}\right|_{x,y} \tag{A.3.16}$$

Where:

$$\left.\frac{\partial f}{\partial}\right|_{g_i} = \left[\frac{g_i(2)}{{g_i(1)}^2+{g_i(2)}^2}\middle|\frac{-g_i(1)}{{g_i(1)}^2+{g_i(2)}^2}\right] \tag{A.3.17}$$

$$\left.\frac{\partial g_1}{\partial}\right| = \begin{pmatrix}1&0&0\\0&0&1\end{pmatrix} \tag{A.3.18}$$

$$\left.\frac{\partial g_2}{\partial}\right|_h = \begin{pmatrix}0&-1&0\\\frac{h(1)}{g_2(2)}&0&\frac{h(3)}{g_2(2)}\end{pmatrix} \tag{A.3.19}$$

$$\left.\frac{\partial h}{\partial\Phi_i^e}\right|_{\Phi_i^e,\Phi_i^c,\mathtt{x}} = \Phi_i^e \cdot \left[\frac{\partial R(\phi)}{\partial\phi(1)}|_0 \;\middle|\; \frac{\partial R(\phi)}{\partial\phi(2)}|_0 \;\middle|\; \frac{\partial R(\phi)}{\partial\phi(3)}|_0\right] \cdot \Phi_i^c \cdot \mathtt{x} \tag{A.3.20}$$

$$\left.\frac{\partial h}{\partial\Phi_i^c}\right|_{\Phi_i^e,\Phi_i^c,\mathtt{x}} = \Phi_i^e \cdot \left[\frac{\partial R(\phi)}{\partial\phi(1)}|_0 \;\middle|\; \frac{\partial R(\phi)}{\partial\phi(2)}|_0 \;\middle|\; \frac{\partial R(\phi)}{\partial\phi(3)}|_0\right]^T \cdot \Phi_i^{cT} \cdot \mathtt{x} \tag{A.3.21}$$

$$\left.\frac{\partial h}{\partial\mathtt{x}}\right|_{\mathtt{x}} = \Phi_i^e \cdot \Phi_i^{cT} \tag{A.3.22}$$

$$\left.\frac{\partial\mathtt{x}}{\partial}\right|_{x,y} \overset{*}{=} \frac{b}{d} \cdot \left(e_1^3 - \frac{1}{b}\cdot\mathtt{x} \;\middle|\; \tfrac{1}{2}\cdot e_2^3 \;\middle|\; \frac{1}{b}\cdot\mathtt{x} \;\middle|\; \tfrac{1}{2}\cdot e_2^3\right) \tag{A.3.23}$$

Where the equality $(*)$ is determined in the same way as in (A.3.5) and (A.3.6). Using the above definition of $\mathtt{x}\,(x,y)$ the inverse depth can be

determined by (see (A.2.4), page 219):

$$\rho \quad = \quad |\mathtt{x}\,(x,y)|_2^{-1} \tag{A.3.24}$$

Thus, the Jacobians for inverse depth $\rho$ by $x$ and $y$ is ($\rho$ is the local camera-point distant and thus independent of navigation entities):

$$\begin{aligned}
\frac{\partial \rho}{\partial x,y} &= \left.\frac{\partial\,|\cdot|_2^{-1}}{\partial}\right|_{\mathtt{x}} \cdot \left.\frac{\partial \mathtt{x}}{\partial}\right|_{x,y} = \left.\frac{\partial\,(\cdot)^{-1}}{\partial}\right|_{|\mathtt{x}|_2} \cdot \left.\frac{\partial\,|\cdot|_2}{\partial}\right|_{\mathtt{x}} \cdot \left.\frac{\partial \mathtt{x}}{\partial}\right|_{x,y} \\
&= \left(-\,|\mathtt{x}|_2^{-2}\right) \cdot \left(\frac{\mathtt{x}(1)}{|\mathtt{x}|_2} \;\middle|\; \frac{\mathtt{x}(2)}{|\mathtt{x}|_2} \;\middle|\; \frac{\mathtt{x}(3)}{|\mathtt{x}|_2}\right) \cdot \left.\frac{\partial \mathtt{x}}{\partial}\right|_{x,y} \\
&= -\left(\frac{\mathtt{x}(1)}{|\mathtt{x}|_2^3} \;\middle|\; \frac{\mathtt{x}(2)}{|\mathtt{x}|_2^3} \;\middle|\; \frac{\mathtt{x}(3)}{|\mathtt{x}|_2^3}\right) \cdot \frac{b}{d} \cdot \left(e_1^3 - \frac{1}{b}\cdot \mathtt{x} \;\middle|\; \tfrac{1}{2}\cdot e_2^3 \;\middle|\; \frac{1}{b}\cdot \mathtt{x} \;\middle|\; \tfrac{1}{2}\cdot e_2^3\right) \\
&= -\frac{b}{d\cdot |\mathtt{x}|_2^3} \cdot \mathtt{x}^T \cdot \left(e_1^3 - \frac{1}{b}\cdot \mathtt{x} \;\middle|\; \tfrac{1}{2}\cdot e_2^3 \;\middle|\; \frac{1}{b}\cdot \mathtt{x} \;\middle|\; \tfrac{1}{2}\cdot e_2^3\right) \\
&= -\frac{b}{d\cdot |\mathtt{x}|_2^3} \cdot \left(\mathtt{x}(1) - \frac{|\mathtt{x}|_2^2}{b} \;\middle|\; \tfrac{1}{2}\cdot \mathtt{x}(2) \;\middle|\; \frac{|\mathtt{x}|_2^2}{b} \;\middle|\; \tfrac{1}{2}\cdot \mathtt{x}(2)\right)
\end{aligned} \tag{A.3.25}$$

**Hessian** For the analysis of covariance propagation the first component's Hessian for $sphere\,(t_{cX}^e)$ ($\phi$, see equation (4.2.3), page 70) is needed. Since the analysis is done using synthetic data, without taking $\Gamma_i^c$ into account, the latter is neglected in equation (A.3.12). Moreover, $\mathtt{x}$ is simplified to $(x(1), x(2), 1)$. Let $p = (\Phi_i^e, x(1), x(2))$. Due to the scale invariance of $\arctan_2$ equation (4.2.3) it's Jacobian can be computed by:

$$\begin{aligned}
\phi \quad &= \quad (f \circ g_1)\,(h_{(\Phi_i^e, x(1), x(2))}) \quad = \quad (f \circ g_1)\,(h_{(p)}) \\
\Longrightarrow \left.\frac{\partial \phi}{\partial}\right|_p \quad &= \quad \left.\frac{\partial f}{\partial}\right|_{g_1} \cdot \left.\frac{\partial g_1}{\partial}\right|_h \cdot \left.\frac{\partial h}{\partial}\right|_p \\
&= \quad \begin{pmatrix} \frac{h(3)}{h(1)^2+h(3)^2} & \frac{-h(1)}{h(1)^2+h(3)^2} \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \left.\frac{\partial h}{\partial}\right|_p
\end{aligned}$$

Let $\eta := {h_{(1)}}^2 + {h_{(3)}}^2$. By this the Hessian given by:

$$\begin{aligned}\frac{\partial^2 (f \circ g_1)\,(h_{(p)})}{\partial p_{\mathtt{i}} \partial p_{\mathtt{j}}} &= \frac{\partial \begin{pmatrix} \frac{h_{(3)}}{\eta} & \frac{-h_{(1)}}{\eta} \end{pmatrix}}{\partial p_{\mathtt{j}}} \cdot \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \frac{\partial h}{\partial p_{\mathtt{i}}} \\ &\quad + \begin{pmatrix} \frac{h_{(3)}}{\eta} & \frac{-h_{(1)}}{\eta} \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \frac{\partial^2 h}{\partial p_{\mathtt{i}} \partial p_{\mathtt{j}}} \quad \text{(A.3.26)}\end{aligned}$$

Where:

$$\begin{aligned}\frac{\partial \frac{h_{(3)}}{\eta}}{\partial p_{\mathtt{j}}} &= \frac{1}{\eta^2} \cdot \left( \frac{\partial h_{(3)}}{\partial p_{\mathtt{j}}} \cdot \eta - h_{(3)} \cdot \frac{\partial \eta}{\partial p_{\mathtt{j}}} \right) \\ &= \frac{1}{\eta^2} \cdot \left( \frac{\partial h_{(3)}}{\partial p_{\mathtt{j}}} \cdot \eta - h_{(3)} \cdot \left( 2h_{(1)} \frac{\partial h_{(1)}}{\partial p_{\mathtt{j}}} + 2h_{(3)} \frac{\partial h_{(3)}}{\partial p_{\mathtt{j}}} \right) \right) \\ &= \frac{1}{\eta^2} \cdot \left( \frac{\partial h_{(3)}}{\partial p_{\mathtt{j}}} \cdot \eta - 2h_{(1)} h_{(3)} \frac{\partial h_{(1)}}{\partial p_{\mathtt{j}}} - 2{h_{(3)}}^2 \frac{\partial h_{(3)}}{\partial p_{\mathtt{j}}} \right) \\ &= \frac{1}{\eta^2} \cdot \left( \left( {h_{(1)}}^2 - {h_{(3)}}^2 \right) \frac{\partial h_{(3)}}{\partial p_{\mathtt{j}}} - 2h_{(1)} h_{(3)} \frac{\partial h_{(1)}}{\partial p_{\mathtt{j}}} \right) \quad \text{(A.3.27)} \\ \frac{\partial \frac{-h_{(1)}}{\eta}}{\partial p_{\mathtt{j}}} &= \frac{1}{\eta^2} \cdot \left( -\frac{\partial h_{(1)}}{\partial p_{\mathtt{j}}} \cdot \eta + h_{(1)} \cdot \frac{\partial \eta}{\partial p_{\mathtt{j}}} \right) \\ &\overset{\sim \text{(A.3.27)}}{=} \frac{1}{\eta^2} \cdot \left( \left( {h_{(1)}}^2 - {h_{(3)}}^2 \right) \frac{\partial h_{(1)}}{\partial p_{\mathtt{j}}} + 2h_{(1)} h_{(3)} \frac{\partial h_{(3)}}{\partial p_{\mathtt{j}}} \right) \quad \text{(A.3.28)}\end{aligned}$$

The derivatives $\frac{\partial h}{\partial p_{\mathtt{j}}}$ can be determined as above, using $\mathbf{x} = \begin{pmatrix} x_{(1)} & x_{(2)} & 1 \end{pmatrix}$. The second derivatives can be computed straight forward from $h\,(p) = \Phi_i^e \cdot \begin{pmatrix} x_{(1)} & x_{(2)} & 1 \end{pmatrix}^T$.

## A.3.2 Observation Models

For the derivatives of the homogeneous projections $\mathbf{x}$ and $\mathbf{y}$ of the landmark $\mathcal{X}$ for ES, ID and PD respectively the following conventions are used. These identities ease the readability of the following derivations.

$$\begin{pmatrix} h_c\,(\mathcal{X}) \\ h_{c'}\,(\mathcal{X}) \end{pmatrix} := \begin{pmatrix} \mathbf{H}\,(\mathbf{x}) \\ \mathbf{H}\,(\mathbf{y}) \end{pmatrix} = h\,(\mathcal{X}) \qquad \text{(A.3.29)}$$

$$P_{12} := \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \text{ and } P_3 := \begin{pmatrix} 0 & 0 & 1 \end{pmatrix} \tag{A.3.30}$$

Moreover, let $\mathtt{x}_{(\mathcal{X})}$ and $\mathtt{y}_{(\mathcal{X})}$ the representatives of $\mathbf{x}_{(\mathcal{X})}$ and $\mathbf{y}_{(\mathcal{X})}$ as determined in appendix A.2.2, pages 219ff (see equations (A.2.8) - (A.2.17)). It holds:

$$h_c(\mathcal{X}) = \mathbf{H}(\mathbf{x}) = \frac{P_{12} \cdot \mathtt{x}_{(\mathcal{X})}}{P_3 \cdot \mathtt{x}_{(\mathcal{X})}} \tag{A.3.31}$$

$$h_{c'}(\mathcal{X}) = \mathbf{H}(\mathbf{y}) = \frac{P_{12} \cdot \mathtt{y}_{(\mathcal{X})}}{P_3 \cdot \mathtt{y}_{(\mathcal{X})}} \tag{A.3.32}$$

In the following, $s$ is defined as 1, $\rho$ and $d$ for ES, ID and PD respectively. By this, the Jacobians for $h_c$ and $h_{c'}$ can be determined using the Jacobian of $\mathtt{x}_{(\mathcal{X})}$ and $\mathtt{y}_{(\mathcal{X})}$ by:

$$\begin{aligned}
\frac{\partial h_c}{\partial} &= \frac{\partial \left( P_{12} \cdot \mathtt{x} \cdot (P_3 \cdot \mathtt{x})^{-1} \right)}{\partial} \\
&= P_{12} \cdot \frac{\frac{\partial \mathtt{x}}{\partial} \cdot (P_3 \cdot \mathtt{x}) - \mathtt{x} \cdot P_3 \cdot \frac{\partial \mathtt{x}}{\partial}}{(P_3 \cdot \mathtt{x})^2} \\
&= P_{12} \cdot \frac{(P_3 \cdot \mathtt{x}) \cdot \frac{\partial \mathtt{x}}{\partial} - \mathtt{x} \cdot P_3 \cdot \frac{\partial \mathtt{x}}{\partial}}{(P_3 \cdot \mathtt{x})^2} \\
&= \frac{P_{12}}{(P_3 \cdot \mathtt{x})^2} \cdot \left( (P_3 \cdot \mathtt{x}) \cdot 1^3 - \mathtt{x} \cdot P_3 \right) \cdot \frac{\partial \mathtt{x}}{\partial} \\
&= \frac{P_{12}}{(P_3 \cdot \mathtt{x})^2} \cdot \begin{pmatrix} \mathtt{x}_{(3)} & 0 & -\mathtt{x}_{(1)} \\ 0 & \mathtt{x}_{(3)} & -\mathtt{x}_{(2)} \\ 0 & 0 & \mathtt{x}_{(3)} - \mathtt{x}_{(3)} \end{pmatrix} \cdot \frac{\partial \mathtt{x}}{\partial}
\end{aligned} \tag{A.3.33}$$

$$\begin{aligned}
\frac{\partial s \cdot b \cdot (P_3 \cdot \mathtt{x})^{-1}}{\partial} &= \frac{\frac{\partial s \cdot b}{\partial} \cdot (P_3 \cdot \mathtt{x}) - s \cdot b \cdot \frac{\partial (P_3 \cdot \mathtt{x})}{\partial}}{(P_3 \cdot \mathtt{x})^2} \\
&= \frac{b}{\mathtt{x}_{(3)}{}^2} \cdot \left( \frac{\partial s}{\partial} \cdot \mathtt{x}_{(3)} - s \cdot P_3 \cdot \frac{\partial \mathtt{x}}{\partial} \right)
\end{aligned} \tag{A.3.34}$$

Where $\frac{\partial d}{\partial d} = \frac{\partial \rho}{\partial \rho} = 1$ and $\frac{\partial s}{\partial} = 0$. These identities simplify to:

$$\frac{h_c(\mathcal{X})}{\partial} = \frac{1}{(\mathbf{x}(3))^2} \cdot \begin{pmatrix} \mathbf{x}(3) & 0 & -\mathbf{x}(1) \\ 0 & \mathbf{x}(3) & -\mathbf{x}(2) \end{pmatrix} \cdot \frac{\partial \mathbf{x}}{\partial} \tag{A.3.35}$$

$$\frac{h_{c'}(\mathcal{X})}{\partial} = \frac{h_c(\mathcal{X})}{\partial} - \begin{pmatrix} \frac{b}{\mathbf{x}(3)^2} \cdot \left( \frac{\partial s}{\partial} \cdot \mathbf{x}(3) - s \cdot P_3 \cdot \frac{\partial \mathbf{x}}{\partial} \right) \\ 0 \end{pmatrix} \tag{A.3.36}$$

Using these derivative templates the Jacobians for the observation models, given in appendix A.2.2, can determined.

#### ES observation model

The observation model is given by equation (A.2.8), page 219.

$$\mathbf{x}^{\text{ES}} = \left[ \Phi_i^c \cdot \Phi_i^{eT} \cdot (\mathcal{X}^{\text{ES}} - t_{ei}^e) + t_{ci}^c \right] = [\mathbf{x}^{\text{ES}}]$$

Thus, the derivatives by $\Gamma_i^e$, $\Gamma_c^i$ and $\mathcal{X}^{\text{ES}}$ are (others $\equiv 0$):

$$\frac{\partial \mathbf{x}^{\text{ES}}}{\partial t_{ei}^e} = -\Phi_i^c \cdot \Phi_i^{eT} \tag{A.3.37}$$

$$\frac{\partial \mathbf{x}^{\text{ES}}}{\partial \Phi_i^e} = \Phi_i^c \cdot \left[ \frac{\partial R(\phi)}{\partial \phi(1)}|_0 \;\middle|\; \frac{\partial R(\phi)}{\partial \phi(2)}|_0 \;\middle|\; \frac{\partial R(\phi)}{\partial \phi(3)}|_0 \right]^T \cdot \Phi_i^{eT} \cdot (\mathcal{X}^{\text{ES}} - t_{ei}^e) \tag{A.3.38}$$

$$\frac{\partial \mathbf{x}^{\text{ES}}}{\partial t_{ci}^c} = 1^3 \tag{A.3.39}$$

$$\frac{\partial \mathbf{x}^{\text{ES}}}{\partial \Phi_i^c} = \Phi_i^c \cdot \left[ \frac{\partial R(\phi)}{\partial \phi(1)}|_0 \;\middle|\; \frac{\partial R(\phi)}{\partial \phi(2)}|_0 \;\middle|\; \frac{\partial R(\phi)}{\partial \phi(3)}|_0 \right] \cdot \Phi_i^{eT} \cdot (\mathcal{X}^{\text{ES}} - t_{ei}^e) \tag{A.3.40}$$

$$\frac{\partial \mathbf{x}^{\text{ES}}}{\partial \mathcal{X}^{\text{ES}}} = \Phi_i^c \cdot \Phi_i^{eT} \tag{A.3.41}$$

#### ID observation model

The observation model is given by equation (A.2.10), page 220.

$$\mathbf{x}^{\text{ID}} = \left[ \Phi_i^c \cdot \Phi_i^{eT} \cdot (ray(\phi, \psi) + \rho \cdot (\mathcal{C} - t_{ei}^e)) + \rho \cdot t_{ci}^c \right] = [\mathbf{x}^{\text{ID}}]$$

$$g \quad := \quad (ray\,(\phi,\psi) + \rho{\cdot}(\mathcal{C} - t^e_{ei}))$$

Thus, the derivatives by $\pi$, $\phi$ and $\mathcal{X}^{\text{ID}} = \left(c^T,\ \phi,\ \psi,\ \rho\right)^T$ are (others $\equiv 0$):

$$\frac{\partial \mathbf{x}^{\text{ID}}}{\partial t^e_{ei}} \quad = \quad \Phi^c_i{\cdot}\Phi^{eT}_i{\cdot}\left(-\rho{\cdot}1^3\right) \quad = \quad -\rho{\cdot}\Phi^c_i{\cdot}\Phi^{eT}_i \tag{A.3.42}$$

$$\frac{\partial \mathbf{x}^{\text{ID}}}{\partial \Phi^e_i} \quad = \quad \Phi^c_i{\cdot}\left[\,\tfrac{\partial R(\phi)}{\partial\phi(1)}|_0 \;\middle|\; \tfrac{\partial R(\phi)}{\partial\phi(2)}|_0 \;\middle|\; \tfrac{\partial R(\phi)}{\partial\phi(3)}|_0\right]^T{\cdot}\Phi^{eT}_i{\cdot}g \tag{A.3.43}$$

$$\frac{\partial \mathbf{x}^{\text{ID}}}{\partial t^c_{ci}} \quad = \quad \rho{\cdot}1^3 \tag{A.3.44}$$

$$\frac{\partial \mathbf{x}^{\text{ID}}}{\partial \Phi^c_i} \quad = \quad \Phi^c_i{\cdot}\left[\,\tfrac{\partial R(\phi)}{\partial\phi(1)}|_0 \;\middle|\; \tfrac{\partial R(\phi)}{\partial\phi(2)}|_0 \;\middle|\; \tfrac{\partial R(\phi)}{\partial\phi(3)}|_0\right]{\cdot}\Phi^{eT}_i{\cdot}g \tag{A.3.45}$$

$$\frac{\partial \mathbf{x}^{\text{ID}}}{\partial \mathcal{C}} \quad = \quad \rho{\cdot}\Phi^c_i{\cdot}\Phi^{eT}_i \tag{A.3.46}$$

$$\begin{aligned}\frac{\partial \mathbf{x}^{\text{ID}}}{\partial \phi} \quad &= \quad \Phi^c_i{\cdot}\Phi^{eT}_i{\cdot}\frac{\partial ray\,(\phi,\psi)}{\partial\phi} \\ &\overset{(4.2.27)}{=} \quad \Phi^c_i{\cdot}\Phi^{eT}_i{\cdot}\begin{pmatrix}\cos(\phi){\cdot}\cos(\psi)\\ 0\\ -\sin(\phi){\cdot}\cos(\psi)\end{pmatrix}\end{aligned} \tag{A.3.47}$$

$$\begin{aligned}\frac{\partial \mathbf{x}^{\text{ID}}}{\partial \psi} \quad &= \quad \Phi^c_i{\cdot}\Phi^{eT}_i{\cdot}\frac{\partial ray\,(\phi,\psi)}{\partial\psi} \\ &\overset{(4.2.27)}{=} \quad \Phi^c_i{\cdot}\Phi^{eT}_i{\cdot}\begin{pmatrix}-\sin(\phi){\cdot}\sin(\psi)\\ -\cos(\psi)\\ -\cos(\phi){\cdot}\sin(\psi)\end{pmatrix}\end{aligned} \tag{A.3.48}$$

$$\frac{\partial \mathbf{x}^{\text{ID}}}{\partial \rho} \quad = \quad \Phi^c_i{\cdot}\Phi^{eT}_i{\cdot}\left(\mathcal{C} - t^i_{ei}\right) + t^c_{ci} \tag{A.3.49}$$

#### PD observation model

Using $\mathcal{X}^{\text{PD}} = (x^{\text{PD}},\ d)$ equation (A.2.16) on page 221 results in:

$$\mathbf{x}^{\text{PD}} \quad = \quad \left[\Phi^c_i{\cdot}\Phi^{nT}_i{\cdot}\left(\Phi^i_c{\cdot}\left(b{\cdot}\begin{pmatrix}x^{\text{PD}}\\ 1\end{pmatrix} - d{\cdot}t^c_{ci}\right) - d{\cdot}t^n_{ni}\right) + d{\cdot}t^c_{ci}\right] \quad = \quad [\mathbf{x}^{\text{PD}}]$$

In the following, let:

$$f \;:=\; b\cdot\left(\begin{smallmatrix} x^{\mathrm{PD}} \\ 1 \end{smallmatrix}\right) - d\cdot t^c_{ci} \qquad \text{and} \qquad g \;:=\; \Phi_i^{cT}\cdot f - d\cdot t^n_{ni}$$

By this the derivatives by $\Gamma^n_i$, $\Gamma^c_i$ and $\mathcal{X}^{\mathrm{PD}}$ are (others $\equiv 0$):

$$\frac{\partial \mathtt{x}^{\mathrm{PD}}}{\partial t^n_{ni}} = -d\cdot\Phi^c_i\cdot\Phi_i^{nT} \tag{A.3.50}$$

$$\frac{\partial \mathtt{x}^{\mathrm{PD}}}{\partial \Phi^n_i} = \Phi^c_i\cdot\left[\frac{\partial R(\phi)}{\partial\phi(1)}|_0 \;\middle|\; \frac{\partial R(\phi)}{\partial\phi(2)}|_0 \;\middle|\; \frac{\partial R(\phi)}{\partial\phi(3)}|_0\right]^T\cdot\Phi_i^{nT}\cdot g \tag{A.3.51}$$

$$\begin{aligned}\frac{\partial \mathtt{x}^{\mathrm{PD}}}{\partial t^c_{ci}} &= -d\cdot\Phi^c_i\cdot\Phi_i^{nT}\cdot\Phi_i^{cT} + d\cdot 1^3 \\ &= d\cdot\left(1^3 - \Phi^c_i\cdot\Phi_i^{nT}\cdot\Phi_i^{cT}\right)\end{aligned} \tag{A.3.52}$$

$$\begin{aligned}\frac{\partial \mathtt{x}^{\mathrm{PD}}}{\partial \Phi^c_i} &= \Phi^c_i\cdot\left[\frac{\partial R(\phi)}{\partial\phi(1)}|_0 \;\middle|\; \frac{\partial R(\phi)}{\partial\phi(2)}|_0 \;\middle|\; \frac{\partial R(\phi)}{\partial\phi(3)}|_0\right]\cdot\Phi_i^{nT}\cdot g \\ &\quad + \Phi^c_i\cdot\Phi_i^{nT}\cdot\left[\frac{\partial R(\phi)}{\partial\phi(1)}|_0 \;\middle|\; \frac{\partial R(\phi)}{\partial\phi(2)}|_0 \;\middle|\; \frac{\partial R(\phi)}{\partial\phi(3)}|_0\right]^T\cdot\Phi_i^{cT}\cdot f\end{aligned} \tag{A.3.53}$$

$$\frac{\partial \mathtt{x}^{\mathrm{PD}}}{\partial x^{\mathrm{PD}}} = b\cdot\Phi^c_i\cdot\Phi_i^{nT}\cdot\Phi_i^{cT}\cdot\left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{smallmatrix}\right) \tag{A.3.54}$$

$$\begin{aligned}\frac{\partial \mathtt{x}^{\mathrm{PD}}}{\partial d} &= \Phi^c_i\cdot\Phi_i^{nT}\cdot\left(-\Phi_i^{cT}\cdot t^c_{ci} - t^n_{ni}\right) + t^c_{ci} \\ &= t^c_{ci} - \Phi^c_i\cdot\Phi_i^{nT}\cdot\left(\Phi_i^{cT}\cdot t^c_{ci} + t^n_{ni}\right)\end{aligned} \tag{A.3.55}$$

As can be seen in equation (A.3.54) the derivative of the predicted pixel position $\mathtt{x}$ by the view direction $x^{\mathrm{PD}}$ in PD is independent of $x^{\mathrm{PD}}$, i. e., it is strictly linear. In contrast, the ID view directions derivative for $\phi, \psi$ (equations (A.3.47), (A.3.48)), is dependent on $\sin, \cos$ of $\psi, \phi$. By this it is not linear.

**Relocalization Models** For the PD relocalization (see section 4.2.7) the actual local pose is added to the old global pose. Moreover, the PD landmarks are transferred to the new local navigation frame. For the pose concatenation the Jacobian matrices are computed straight forward for rotation multiplication and position addition. The models for the PD landmark relocalization in

$x^{\mathrm{PD}}$ are the same as for the observation (see equation (4.2.46), page 86). Thus, the Jacobians given above apply.

The prediction of the new disparity (see equation (4.2.47), page 86) is given by:

$$d \quad = \quad \frac{d_o \cdot b}{P_3 \cdot {\Phi_c^n}^T \cdot \left( b \cdot \begin{pmatrix} x_o^{\mathrm{PD}} \\ 1 \end{pmatrix} - d_o \cdot t_{nc}^n \right)} \quad = \quad P_1 \cdot \left( \mathbf{H}\left( \mathbf{x}_o^{\mathrm{PD}} \right) - \mathbf{H}\left( \mathbf{y}_o^{\mathrm{PD}} \right) \right)$$

By this, the derivatives can be determined using:

$$\frac{\partial d}{\partial} \quad = \quad P_1 \cdot \left( \frac{h_c\left(\mathcal{X}\right)}{\partial} - \frac{h_c\left(\mathcal{X}\right)}{\partial} \right) \tag{A.3.56}$$

$$= \quad \frac{b}{\mathtt{x}(3)^2} \cdot \left( \frac{\partial d_0}{\partial} \cdot \mathtt{x}(3) - s \cdot P_3 \cdot \frac{\partial \mathtt{x}}{\partial} \right) \tag{A.3.57}$$

This ends up with

$$\frac{\partial d}{\partial p} \quad = \quad \begin{cases} \frac{-b}{\mathtt{x}(3)^2} \cdot d_0 \cdot P_3 \cdot \frac{\partial \mathtt{x}}{\partial} & \text{for } \; p = d_0 \\ \frac{b}{\mathtt{x}(3)^2} \cdot \left( \mathtt{x}(3) - d_0 \cdot P_3 \cdot \frac{\partial \mathtt{x}}{\partial} \right) & \text{else} \end{cases} \tag{A.3.58}$$

Appendix B

# Figures

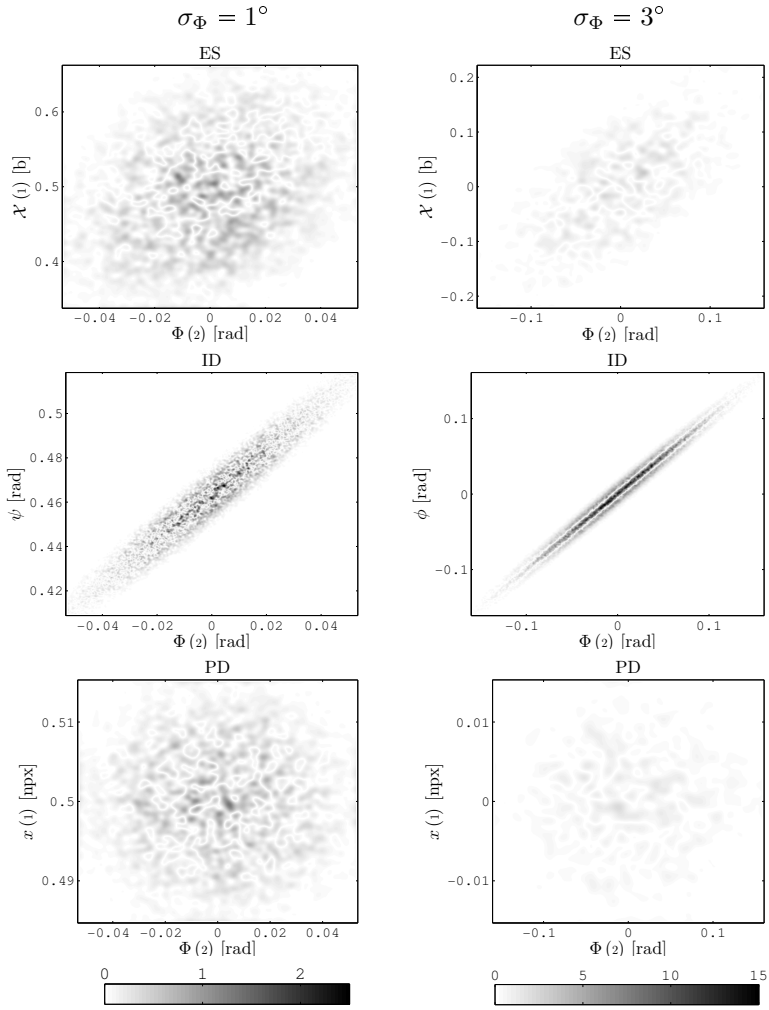


**Figure B.1.** PDF analysis for error propagation of landmark initialization (see section 5.1.3). Error in [%] of propagated PDF compared to landmark particles KDE for $\Phi = 1^3$, $x = 0_2$, $y = (1\ 0)$ (i. e., camera landmark distance 1 baseline). Positions standard deviation 0.05 baselines. First axis represents system heading $\Phi_{(2)}$, second axis for ES: $\mathcal{X}_{(1)}$, ID: $\phi$ and PD: $x^{\mathrm{PD}}{}_{(1)}$. First column was generated with low orientation uncertainty ($1°$), second using medium uncertainty ($3°$).

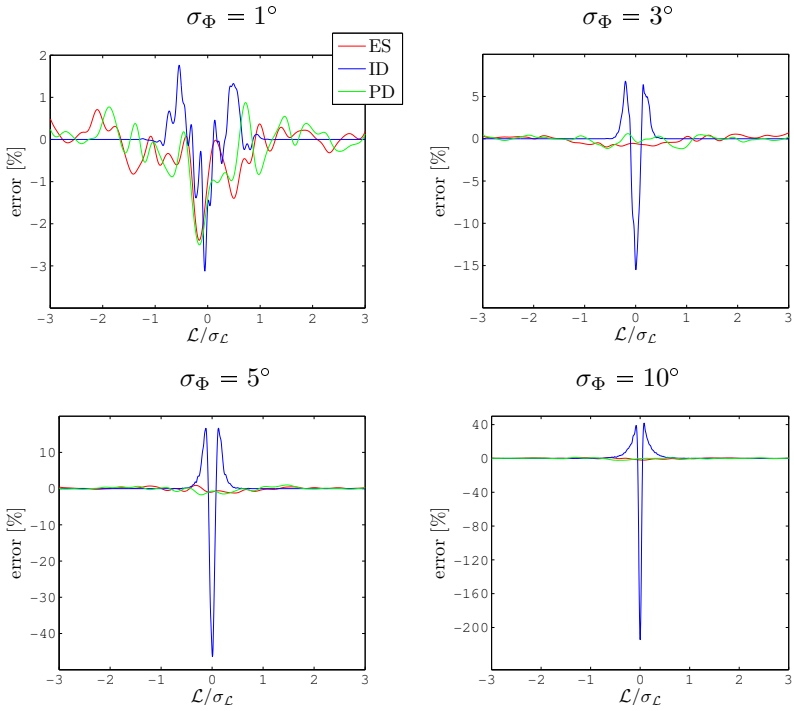


**Figure B.2.** Error of PDFs in percent of maximum density ($max\,(\hat{\varphi}_\mathcal{L})$) for high position uncertainty (standard deviation 0.5 baselines). Horizontal axis represents the first component of landmarks view direction ($\mathcal{L} = \left\{ \mathcal{X}^{\mathrm{ES}}{}_{(1)}\,,\, \phi\,,\, x^{\mathrm{PD}}{}_{(1)} \right\}$) normalized by its standard deviation. PDFs are cross sections for $\Phi_{(2)} = 0$ of distributions for $(\Phi_{(2)}\,,\, \mathcal{L})$. Used initialization setup $\Phi = 1^3$, $x = 0_2$, $y = (1\ 0)$ (i. e., camera landmark distance 1 baseline).

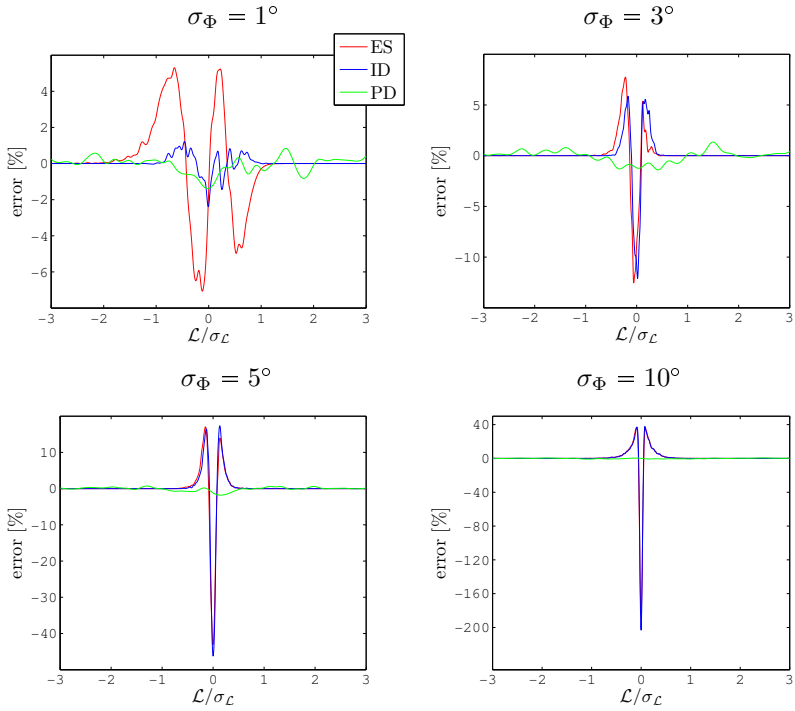


**Figure B.3.** Error of PDFs in percent of maximum density ($max\,(\hat{\varphi}_\mathcal{L})$) for distant landmarks (20 baselines). Horizontal axis represents the first component of landmarks view direction ($\mathcal{L} = \left\{ \mathcal{X}^{\mathrm{ES}}{}_{(1)}\,,\, \phi\,,\, x^{\mathrm{PD}}{}_{(1)} \right\}$) normalized by its standard deviation. PDFs are cross sections for $\Phi_{(2)} = 0$ of distributions for $(\Phi_{(2)}\,,\, \mathcal{L})$. Used initialization setup $\Phi = 1^3$, $x = 0_2$, $y = (0.05\ 0)$. Positions standard deviation 0.05 baselines.

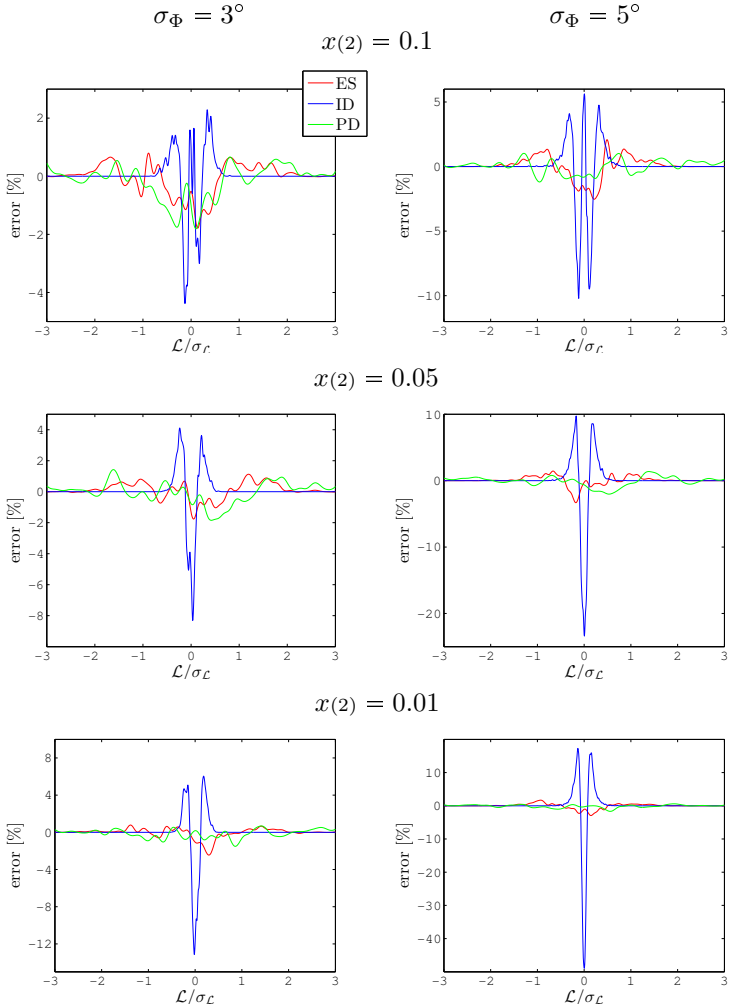


**Figure B.4.** Error of PDFs in percent of maximum density ($max\,(\hat{\varphi}_\mathcal{L})$) for varying pixel heights ($x_{(2)}$) and standard deviations ($\sigma_\Phi$). Horizontal axis: first component of view direction ($\mathcal{L} = \left\{\mathcal{X}^{\mathrm{ES}}{}_{(1)}\,,\,\phi\,,\,x^{\mathrm{PD}}{}_{(1)}\right\}$) normalized by its standard deviation. PDFs are cross sections for $\Phi_{(2)} = 0$ of distributions for $(\Phi_{(2)}\,,\,\mathcal{L})$. Initialization setup $\Phi = 1^3$, $y = (x_{(1)} + 1\,,\,x_{(2)})$ (i. e., landmark distance 1 baseline).

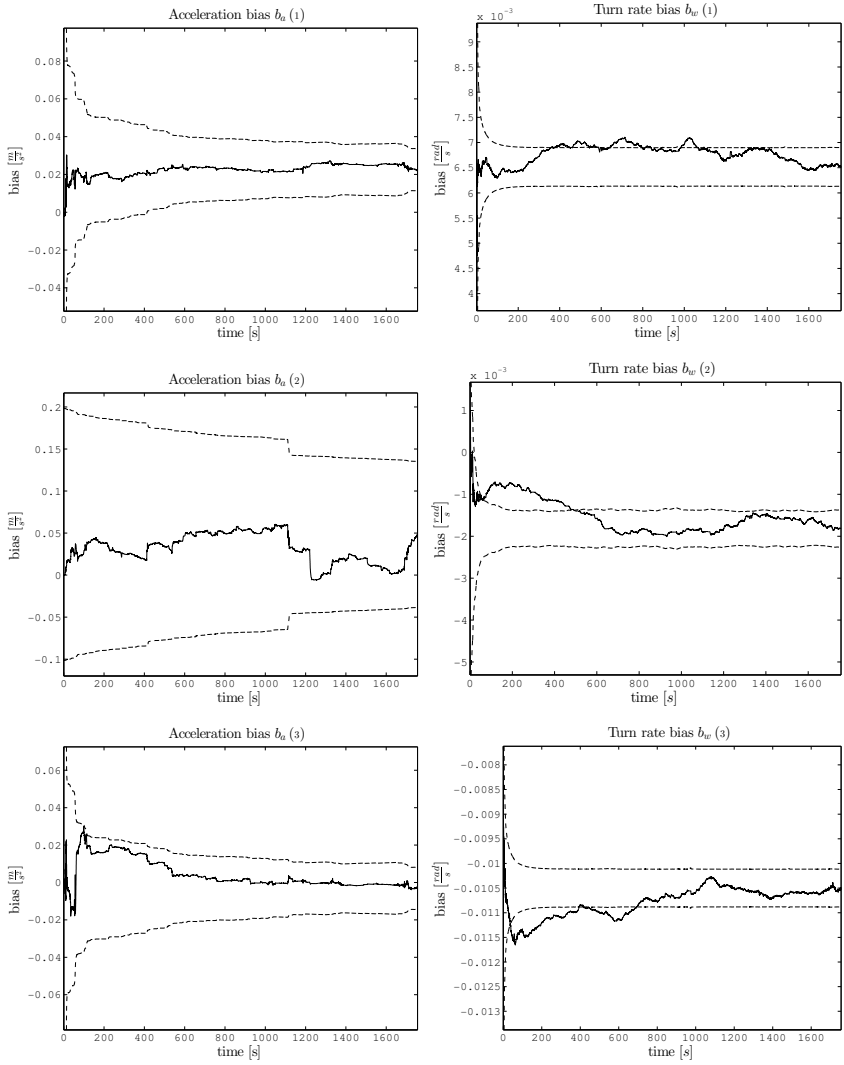


**Figure B.5.** PD parametrization bias estimations for accelerations and turn rates, parameter setup S1. Solid lines: estimated biases; dashed lines: $3\sigma$-bounds.

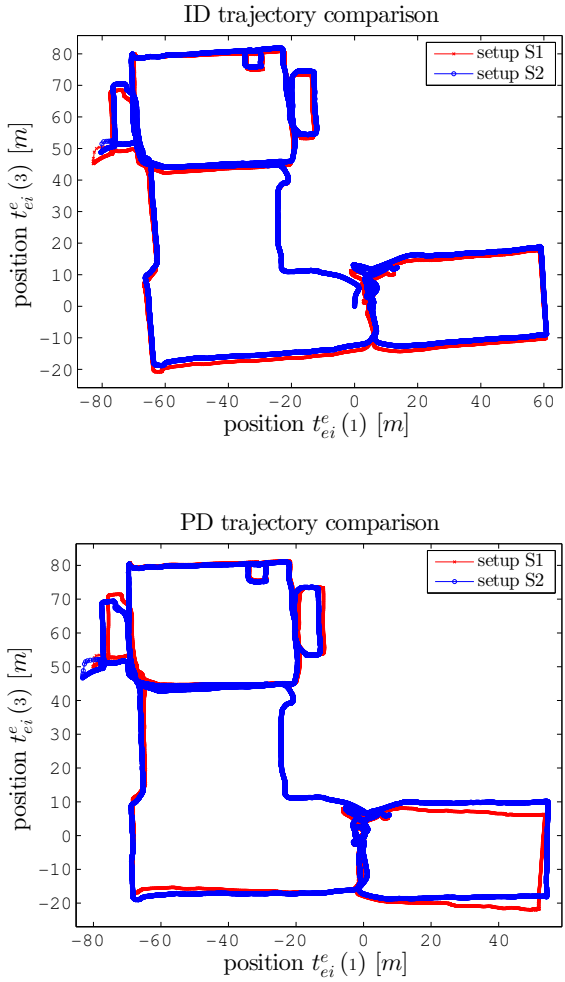


**Figure B.6.** Comparison of estimated trajectories for S1 and S2 using the ID (top) and PD (bottom) estimator.

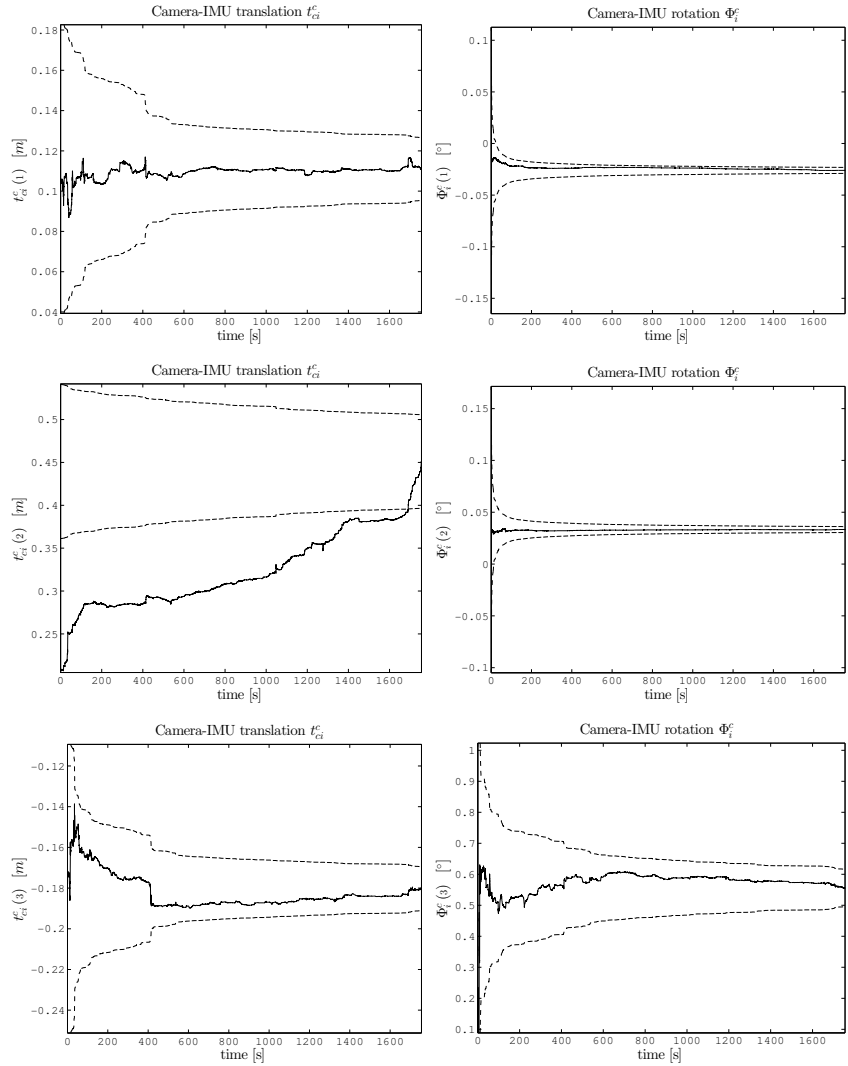


**Figure B.7.** Estimated camera to IMU calibration for dataset S1, ES parametrization. Solid lines: estimated alignment; dashed lines: $3\sigma$-bounds.

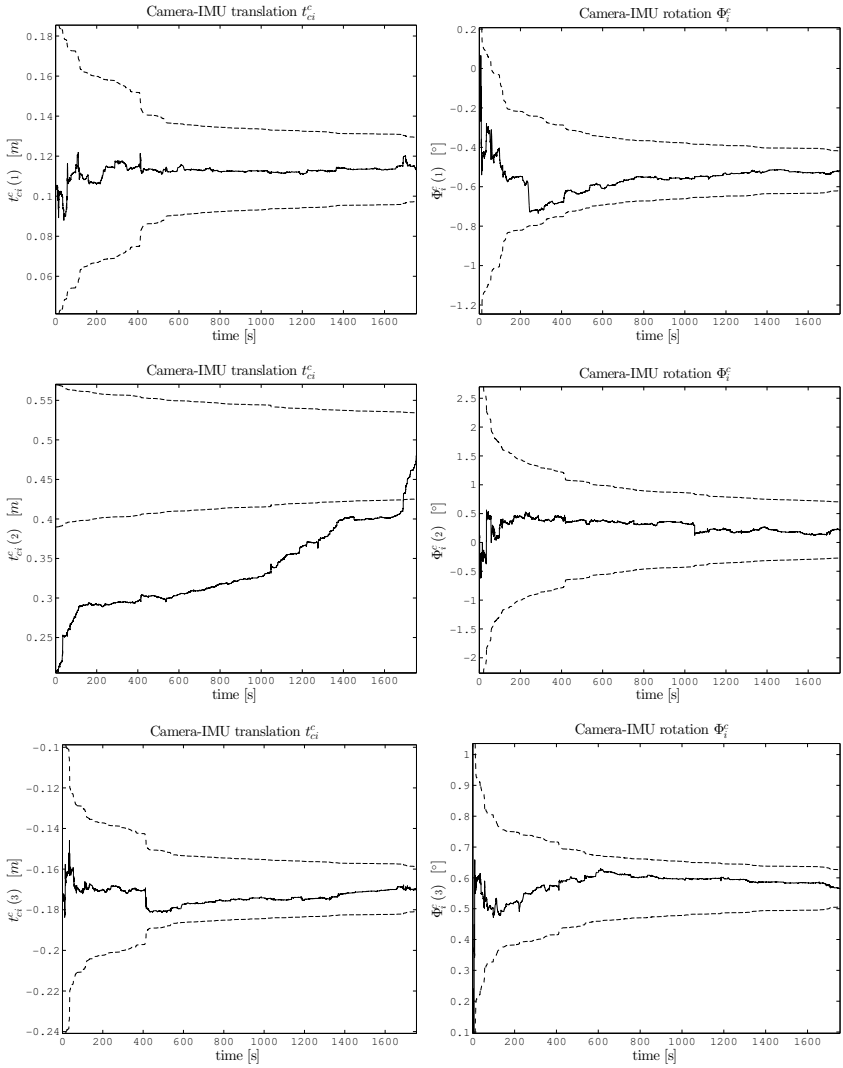


**Figure B.8.** Estimated camera to IMU calibration for dataset S1, ID parametrization. Solid lines: estimated alignment; dashed lines: $3\sigma$-bounds.

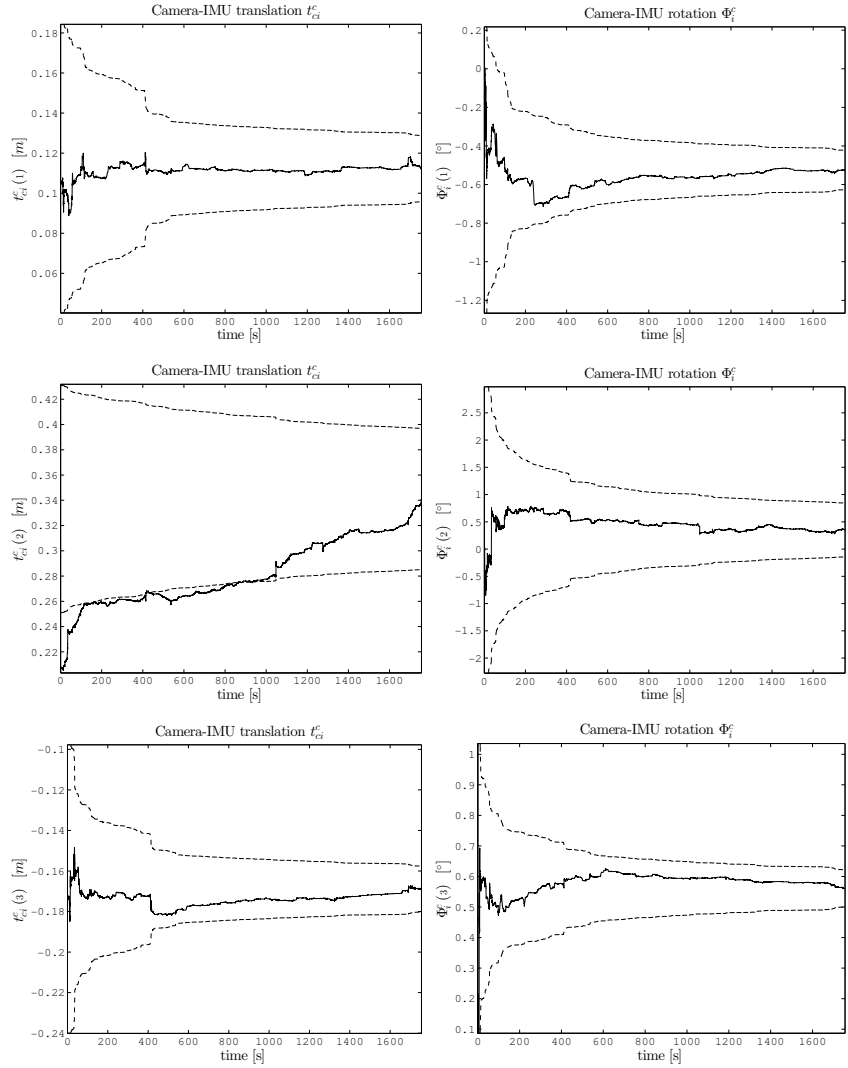


**Figure B.9.** Estimated camera to IMU calibration for dataset S1, PD parametrization. Solid lines: estimated alignment; dashed lines: $3\sigma$-bounds.

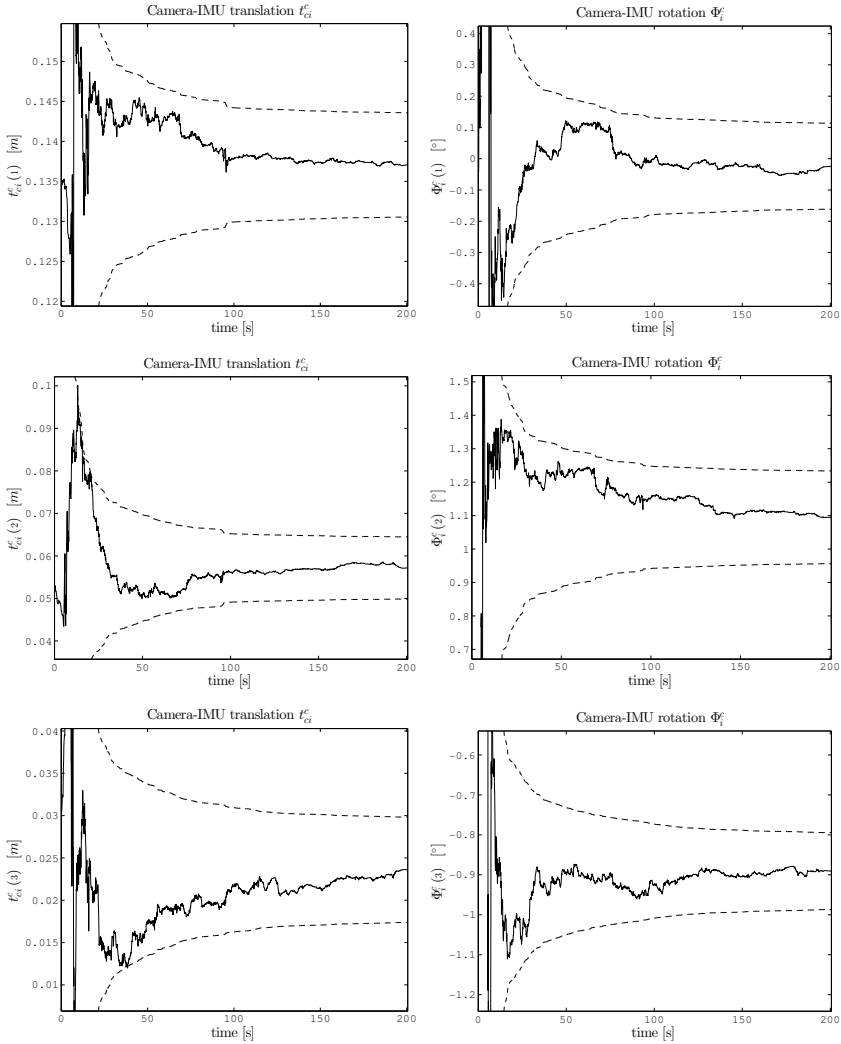


**Figure B.10.** Estimated camera to IMU calibration for online free-hand test run, PD parametrization. Solid lines: estimated alignment; dashed lines: $3\sigma$-bounds.

## Appendix C

# Tables

**Table C.1.** Informativity for submatrices $C'_{\mathrm{ES}}$, $C'_{\mathrm{ID}}$ and $C'_{\mathrm{PD}}$ with respect to generated point cloud (initialized landmarks) for high position uncertainty (standard deviation 0.5 baselines). Used initialization setup $\Phi = 1^3$, $x = 0_2$, $y = (1\ 0)$ (i. e., camera landmark distance 1 baseline).

| | $\mathcal{I}_{0.5}$ | $\mathcal{I}_1$ | $\mathcal{I}_2$ | $\mathcal{I}_3$ | $\mathcal{I}_{0.5}$ | $\mathcal{I}_1$ | $\mathcal{I}_2$ | $\mathcal{I}_3$ |
|---|---|---|---|---|---|---|---|---|
| | $\sigma_\Phi = 1°$ | | | | $\sigma_\Phi = 3°$ | | | |
| ES | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| ID | 0 | 0 | 0 | 0 | **−3** | **−4** | **−3** | −1 |
| PD | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | $\sigma_\Phi = 5°$ | | | | $\sigma_\Phi = 10°$ | | | |
| ES | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| ID | **−11** | **−17** | **−14** | **−7** | **−24** | **−41** | **−48** | **−40** |
| PD | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table C.2.** Informativity for submatrices $C'_{\mathrm{ES}}$, $C'_{\mathrm{ID}}$ and $C'_{\mathrm{PD}}$ for different orientation initialization standard deviations ($\sigma_\Phi$). Camera landmarks distance is 20 baselines. Used initialization setup $\Phi = 1^3$, $x = 0_2$, $y = (0.05\ 0)$. Positions standard deviation 0.05 baselines.

| | $\mathcal{I}_{0.5}$ | $\mathcal{I}_1$ | $\mathcal{I}_2$ | $\mathcal{I}_3$ | $\mathcal{I}_{0.5}$ | $\mathcal{I}_1$ | $\mathcal{I}_2$ | $\mathcal{I}_3$ |
|---|---|---|---|---|---|---|---|---|
| | $\sigma_\Phi = 1°$ | | | | $\sigma_\Phi = 3°$ | | | |
| ES | 0 | −1 | **−3** | −2 | **−4.5** | **−11** | **−13** | **−7.5** |
| ID | 0 | 0 | 0 | 0 | **−3** | **−4** | −2 | −1 |
| PD | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | $\sigma_\Phi = 5°$ | | | | $\sigma_\Phi = 10°$ | | | |
| ES | **−13** | **−25** | **−28** | **−19** | **−26** | **−47** | **−59** | **−52** |
| ID | **−11** | **−16** | **−14** | **−7** | **−24** | **−41** | **−48** | **−40** |
| PD | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table C.3.** Informativity for submatrix $C'_{\mathrm{ID}}$ for varying pixel heights ($x_{(2)} = y_{(2)}$) and orientations standard deviations ($\sigma_\Phi$). Used initialization setup $\Phi = 1^3$, landmark distance 1 baseline and position standard deviation 0.05. In comparison to table 5.3.

| | $\sigma_\Phi = 3°$ | | | | $\sigma_\Phi = 5°$ | | | |
|---|---|---|---|---|---|---|---|---|
| $x_{(2)}$ | $\mathcal{I}_{0.5}$ | $\mathcal{I}_1$ | $\mathcal{I}_2$ | $\mathcal{I}_3$ | $\mathcal{I}_{0.5}$ | $\mathcal{I}_1$ | $\mathcal{I}_2$ | $\mathcal{I}_3$ |
| ±0.01 | **−3** | **−4** | −2 | −1 | **−11** | **−16** | **−14** | **−7** |
| ±0.05 | **−2** | **−3** | −2 | −1 | **−6** | **−9** | **−9** | **−4** |
| ±0.1 | 0 | −1.0 | −2 | −1 | 0 | **−3** | **−5** | **−2** |

**Table C.4.** Informativity of observation prediction, avg. $\mathcal{I}\left(M_{2D}, C^{\mathcal{L}}_{2D,i}\right)$ for predicted master $x_i$ and slave $y_i$ observations with respect to the predicted covariances $C^{\mathcal{L}}_{2D,i}$ for each particle. Escept for the orientations standard deviation of 3°, test setup is the same as in table 5.7, page 118

| | distance 1$[b]$ | | | | distance 25$[b]$ | | | |
|---|---|---|---|---|---|---|---|---|
| | $\mathcal{I}_{0.5}$ | $\mathcal{I}_1$ | $\mathcal{I}_2$ | $\mathcal{I}_3$ | $\mathcal{I}_{0.5}$ | $\mathcal{I}_1$ | $\mathcal{I}_2$ | $\mathcal{I}_3$ |
| ES | 0 | 0 | 0 | 0 | **15** | **16** | **3** | 0 |
| ID | 1 | 1 | 0 | 0 | **5** | **5** | −2 | −1 |
| PD | 0 | 0 | 0 | 0 | 0 | 0 | −2 | −1 |

**Table C.5.** Root mean square error of $\varphi^{\mathcal{L}}_{2D}$ with respect to $\hat{\varphi}_{2D}$ for master $x$ and slave $y$ reprojections. Values given in % of $max\left(\hat{\varphi}_{2D}\right)$. Test setup is the same as in tables 5.7 (page 118) and C.4 (page 247) respectively.

| distance: | 1$[b]$ | | | | 25$[b]$ | | | |
|---|---|---|---|---|---|---|---|---|
| stdev. $\Phi$ = | 1° | | 3° | | 1° | | 3° | |
| | $x$ | $y$ | $x$ | $y$ | $x$ | $y$ | $x$ | $y$ |
| ES | 0% | 0% | 1% | 1% | **3**% | **3**% | **5.3**% | **5.5**% |
| ID | 0% | 0% | 0% | 0% | 0% | 0% | 1% | 1% |
| PD | 0% | 0% | 0% | 0% | 0% | 0% | 0% | 0% |

**Table C.6.** Comparison of plane navigation error analysis for S1: $\Phi_c^i$ high and low initialization error. Root mean square errors, absolute trajectory ATE and relative pose error RPE for estimation with high initialization error. Percentage of error with respect to low initialization error estimation (see table 7.10, page 174).

| | RMSE | | |
|---|---|---|---|
| | $t^e_{ei(1)}$ $[m]$ | $t^e_{ei(3)}$ $[m]$ | $\Phi^e_{i(2)}$ $[^\circ]$ |
| ES | 6.555 | 3.993 | 5.010 |
| | 454% | 239% | 194% |
| ID | 2.45 | 2.783 | 3.539 |
| | 85% | 86% | 94% |
| PD | 2.101 | 1.582 | 1.878 |
| | 101% | 101% | 100% |

| | Rawseeds error | | |
|---|---|---|---|
| | ATE $\bar{\varepsilon} \pm \sigma_\varepsilon$ | T-RPE $[m]$ | R-RPE $[^\circ]$ |
| ES | 6.3588 ± 4.2982 | 1.4942 | 1.9555 |
| | 316% ± 450% | 218% | 219% |
| ID | 3.0121 ± 2.1622 | 0.771 | 0.9115 |
| | 86% ± 84% | 84% | 90% |
| PD | 1.9676 ± 1.7452 | 0.6618 | 0.8475 |
| | 101% ± 101% | 98% | 100% |

**Table C.7.** Comparison of final height $t^e_{ei(2)}$ error analysis for estimated height in setup S1: $\Phi_c^i$ high and low initialization error. Error for height $t^e_{ei(2)}$; $\epsilon$ final error, $3\sigma$ confidence interval and RMSE for floor E1. Error in percentage of S1 with low initialization error (see table 7.11, page 175).

| | $\epsilon$ | $3\sigma$ | RMSE |
|---|---|---|---|
| ES | -73.44 | ± 0.773 | 55.65 |
| | 816% | NA | 825% |
| ID | -2.192 | ± 1.299 | 1.508 |
| | 108% | NA | 112% |
| PD | -1.747 | ± 1.312 | 1.211 |
| | 100% | NA | 99% |

# Bibliography


[Art+09] Jorge Artieda, José M. Sebastian, Pascual Campoy, Juan F. Correa, Iván F. Mondragón, Carol Martínez, and Miguel Olivares. "Visual 3-D SLAM from UAVs". In: *J. Intell. Robotics Syst.* 55.4-5 (Aug. 2009), pp. 299–321. ISSN: 0921-0296. DOI: 10.1007/s10846-008-9304-8.

[Art57] E. Artin. Geometric Algebra. John Wiley & Sons Inc, 1957. ISBN: 0470034327.

[Bai+06] T. Bailey, J. Nieto, J. Guivant, M. Stevens, and E. Nebot. "Consistency of the ekf-slam algorithm". In: *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on.* Oct. 2006, pp. 3562–3568. DOI: 10.1109/IROS.2006.281644.

[Ble09] Gabriele Bleser. "'Towards Visual-Inertial SLAM for Mobile Augmented Reality'". PhD thesis. Technische Universität Kaiserslautern, 2009.

[BM04] S. Baker and I. Matthews. "Lucas-kanade 20 years on: a unifying framework". In: *International Journal of Computer Vision* 56.3 (Feb. 2004), pp. 221–255. ISSN: 0920-5691. DOI: 10.1023/B:VISI.0000011205.11775.fd. URL: http://www.springerlink.com/content/t032h4pg050x2012/.

[Bon+06] Andrea Bonarini, Wolfram Burgard, Giulio Fontana, Matteo Matteucci, Domenico Giorgio Sorrenti, and Juan Domingo Tardos. "Rawseeds: robotics advancement through web-publishing of sensorial and elaborated extensive data sets". In: *proceedings of IROS'06 Workshop on Benchmarks in Robotics Research.* 2006. URL: http://www.robot.uji.es/EURON/en/iros06.htm.

[Bou99] J. Y. Bouguet. "'Visual methods for three-dimensional modelling'". PhD thesis. California Institute of Technology Pasadena, CA, USA: California Institute of Technology, 1999. DOI: http://dx.

doi.org/10.1117/12.441551. URL: http://etd.caltech.edu/etd/available/etd-02072008-115723/.

[Bro85] R. Brooks. "Visual map making for a mobile robot". In: *Proceedings IEEE International Conference on Robotics and Automation 1985*. Vol. 2. Mar. 1985, pp. 824–829. DOI: 10.1109/ROBOT.1985.1087348.

[Bro86] Rodney A. Brooks. "A robust layered control system for a mobile robot". In: *IEEE Journal of Robotics and Automation* RA-2.1 (Mar. 1986).

[BS96] I.N. Bronstein and K.A. Semendjajew. Taschenbuch der Mathematik. B.G. Teubner Verlagsgesellschaft Leipzig, 1996. ISBN: 3-8154-2001-6.

[Bud12] Maximilian Buder. "Dense real-time stereo matching using memory efficient semi-global-matching variant based on fpgas". In: ed. by Nasser Kehtarnavaz and Matthias F. Carlsohn. Vol. 8437. 1. Brussels, Belgium: SPIE, 2012, pages. DOI: 10.1117/12.921147.

[Bur+09] W. Burgard, C. Stachniss, G. Grisetti, B. Steder, R. Kümmerle, C. Dornhege, M. Ruhnke, A. Kleiner, and Juan D. Tardós. "A comparison of slam algorithms based on a graph of relations". In: *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*. St. Louis, MO, USA, Oct. 2009.

[Cas+07] J. A. Castellanos, R. Martínez-Cantín, J. D. Tardós, and J. Neira. "Robocentric map joining: improving the consistency of ekf-slam". In: *Robotics and Autonomous Systems* 55 (2007), pp. 21–29.

[CDM07] J. Civera, A.J. Davison, and J. M M Montiel. "Inverse depth to depth conversion for monocular slam". In: *Robotics and Automation, 2007 IEEE International Conference on*. 2007, pp. 2778–2783. DOI: 10.1109/ROBOT.2007.363892.

[CDM08] J. Civera, A. J. Davison, and J. M. M. Montiel. "Inverse depth parametrization for monocular slam". In: *Autonomous Robots* 24.5 (2008), pp. 932–945.

[Cer+09] Simone Ceriani, Giulio Fontana, Alessandro Giusti, Daniele Marzorati, Matteo Matteucci, Davide Migliore, Davide Rizzi, Domenico G. Sorrenti, and Pierluigi Taddei. "Rawseeds ground truth collection systems for indoor self-localization and mapping". In: *Autonomous Robots* 27.4 (2009), pp. 353–371. DOI: 10.1007/s10514-009-9156-5. URL: http://www.springerlink.com/content/k924032g72818h53/.

[Cer+11] Simone Ceriani, Daniele Marzorati, Matteo Matteucci, Davide Migliore, and Domenico G. Sorrenti. "On feature parameterization for ekf-based monocular slam". In: *18th World Congress of the International Federation of Automatic Control (IFAC)*. 2011, pp. 6829–6834. DOI: 10.3182/20110828-6-IT-1002.02631.

[Cha05] Germain Chartier. Introduction to Optics. Springer, 2005. ISBN: 0-387-40346-9.

[Çin10] Erhan Çinlar. Probability and Stochastics. Graduate Texts in Mathematics. Springer, 2010. ISBN: 978-0-387-87858-4.

[Civ+09] Javier Civera, Oscar G. Grasa, Andrew J. Davison, and J. M. M. Montiel. "1-point ransac for ekf-based structure from motion". In: *Proceedings of the 2009 IEEE/RSJ international conference on Intelligent robots and systems*. IROS'09. St. Louis, MO, USA: IEEE Press, 2009, pp. 3498–3504. ISBN: 978-1-4244-3803-7.

[CN07] Mark Cummins and Paul Newman. "Probabilistic appearance based navigation and loop closing". In: *Proc. IEEE International Conference on Robotics and Automation IEEE Internation Conference on Robotics and Automation*. Rome, Apr. 2007.

[CNT04] J. A. Castellanos, J. Neira, and J. D. Tardós. Limits to the consistency of EKF-based SLAM. Lisbon, Portugal, 2004.

[Cot04] Jackson D. Cothren. "'Reliability in Constrained Gauss-Markov Models: An Analytical and Differential Approach with Applications in Photogrammetry'". PhD thesis. The Ohio State University, 2004.

[CP04] J. Chen and A. Pinz. “Structure and motion by fusion of inertial and vision-based tracking”. In: *Digital Imaging in Media and Education*. Ed. by W. Burger and J. Scharinger. Vol. 179. Schriftenreihe. Proceedings of the $28^{th}$ ÖAGM/AAPR Conference. OCG, 2004, pp. 55–62.

[DB06] M. Donoser and H. Bischof. “Efficient maximally stable extremal region (mser) tracking”. In: *Proceedings of CVPR06*. Vol. 1. 2006, pp. 553–560.

[DH03] Tarn Duong and Martin Hazelton. “Plug-in bandwidth matrices for bivariate kernel density estimation”. In: *Journal of Nonparametric Statistics* 15.1 (2003), pp. 17–30. DOI: 10.1080/10485250306039.

[Dis+01] M. W. M. Gamini Dissanayake, Paul Newman, Steven Clark, Hugh F. Durrant-Whyte, and M. Csorba. “A solution to the simultaneous localization and map building (slam) problem”. In: *Transactions on Robotics and Automation* 17.3 (2001), pp. 229–241.

[Ein12] Garry A. Einicke. Smoothing, Filtering and Prediction: Estimating the past, present and future. InTech, 2012. ISBN: 978-953-307-752-9. DOI: 10.5772/2706.

[ESN06] C. Engels, H. Stewénius, and D. Nistér. “Bundle adjustment rules”. In: *Photogrammetric Computer Vision (PCV)*. ISPRS, Sept. 2006.

[FTV00] Andrea Fusiello, Emanuele Trucco, and Alessandro Verri. “A compact algorithm for rectification of stereo pairs”. In: *Mach. Vis. Appl.* 12.1 (2000), pp. 16–22.

[GA01] Mohinder S. Grewal and Angus P. Andrews. Kalman Filtering:Theory and Practice. 2nd ed. John Wiley & Sons Inc., 2001. ISBN: 0-471-26638-8.

[GFP08] L. Guan, J.-S. Franco, and M. Pollefeys. “3d object reconstruction with heterogeneous sensor data”. In: *4th International Symposium on 3D Data Processing, Visualization and Transmission (3DPVT)*. Atlanta, GA, USA, June 2008. URL: http://iparla.labri.fr/publications/2008/GFP08a.

[HK10] Marc Hildebrandt and Frank Kirchner. IMU-Aided Stereo Visual Odometry for Ground-Tracking AUV Applications. Tech. rep. Underwater Robotics Department, DFKI RIC Bremen, 2010.

[HKD06] Damith C. Herath, K. R. S. Kodagoda, and Gamini Dissanayake. Modeling Errors in Small Baseline Stereo for SLAM. Tech. rep. ARC Centre of Excellence in Autonomous Systems (CAS), 2006.

[HMR08] Guoquan Huang, Anastasios I. Mourikis, and Stergios I. Roumeliotis. "Analysis and improvement of the consistency of extended kalman filter based slam". In: *IEEE Internation Conference on Robotics and Automation*. 2008, pp. 473–479.

[HS88] C. G. Harris and M. Stephens. "A combined corner and edge detector". In: *4th Alvey Vision Conference, Manchester*. 1988, pp. 147–151. URL: http://citeseerx.ist.psu.edu/showciting;jsessionid=0A73077DEAF92438DFBCB0C996AE9174?cid=4516.

[HS97] Janne Heikkila and Olli Silven. "A four-step camera calibration procedure with implicit image correction". In: *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on* (1997), pp. 1106–1112. ISSN: 1063-6919. DOI: http://doi.ieeecomputersociety.org/10.1109/CVPR.1997.609468.

[HSG08] Jeroen Hol, Thomas Schön, and Fredrik Gustafsson. "Relative pose calibration of a spherical camera and an imu". In: *Proceedings of the 7th IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR)*. Cambridge, United Kingdom, Sept. 2008.

[Hyg+04] Emmanuel Hygounenc, Il-Kyun Jung, Philippe Souères, and Simon Lacroix. "The autonomous blimp project of laas-cnrs: achievements in flight control and terrain mapping." In: *I. J. Robotic Res.* 23.4-5 (2004), pp. 473–511. URL: http://dblp.uni-trier.de/db/journals/ijrr/ijrr23.html#HygounencJSL04.

[HZ03] Richard Hartley and Andrew Zisserman. Multiple View Geometry. 2nd ed. Cambirdge University Press, 2003.

[İB09] Evren İmre and Marie-Odile Berger. "A 3-component inverse depth parameterization for particle filter slam". In: *Lecture Notes in Computer Science*. Vol. 5748/2009. Springer, 2009, pp. 1–10.

[IBN09] E. Imre, M.-O. Berger, and N. Noury. "Improved inverse-depth parameterization for monocular simultaneous localization and mapping". In: *Proceedings of the 2009 IEEE international conference on Robotics and Automation*. IEEE Internation Conference on Robotics and Automation. Kobe, Japan: IEEE Press, 2009, pp. 1901–1906. ISBN: 978-1-4244-2788-8. URL: http://dl.acm.org/citation.cfm?id=1703435.1703741.

[Jäh02] Bernd Jähne. Digitale Bildverarbeitung. 5th ed. Springer, 2002. ISBN: 3-540-41260-3.

[JJU04] Simon J. Julier, Jeffrey, and K. Uhlmann. "Unscented filtering and nonlinear estimation". In: *Proceedings of the IEEE*. 2004, pp. 401–422.

[JS11] E. Jones and S. Soatto. "Visual-inertial navigation, mapping and localization: a scalable real-time causal approach". In: *International Journal of Robotics Research* (Jan. 2011).

[JU01] S.J. Julier and J.K. Uhlmann. "Simultaneous localisation and map building using split covariance intersection". In: *Intelligent Robots and Systems, 2001. Proceedings. 2001 IEEE/RSJ International Conference on*. Vol. 3. 2001, 1257–1262 vol.3. DOI: 10.1109/IROS.2001.977155.

[JU97] S.J. Julier and J.K. Uhlmann. "A new extension of the kalman filter to nonlinear systems". In: *11th International Symposium on Aerospace and Defense Sensing (AeroSense), Simulations and Controls*. 1997.

[Kal60] R. E. Kalman. "A new approach to linear filtering and prediction problems". In: *Journal of Basic Engineering* 82 (1960).

[Koc99] Karl-Rudolf Koch. Parameter Estimation and Hypothesis Testing in Linear Models. 2nd ed. Springer, 1999. ISBN: 3-540-65257-4.

[KS09] Jonathan Kelly and Gaurav S. Sukhatme. "'Fast Relative Pose Calibration for Visual and Inertial Sensors'". In: *Experimental Robotics: The Eleventh International Symposium*. Ed. by Oussama Khatib, Vijay Kumar, and George J. Pappas. Vol. 54.

Springer Tracts in Advanced Robotics. Springer, Apr. 2009, pp. 515–524. DOI: 10.1007/978-3-642-00196-3_59.

[LA09] M.I. A. Lourakis and A.A. Argyros. "SBA: A Software Package for Generic Sparse Bundle Adjustment". In: *ACM Trans. Math. Software* 36.1 (2009), pp. 1–30. DOI: http://doi.acm.org/10.1145/1486525.1486527.

[LBD04] Tine Lefebvre, Herman Bruyninckx, and Joris De Schutter. "Kalman filters for non-linear systems: a comparison of performance". In: *International Journal of Control* 77.7 (2004), pp. 639–653. DOI: 10.1080/00207170410001704998.

[LD07] Jorge Lobo and Jorge Dias. "Relative pose calibration between visual and inertial sensors". In: *I. J. Robotic Res.* 26.6 (2007), pp. 561–575.

[LGK11] H. Lategahn, A. Geiger, and B. Kitt. "Visual slam for autonomous ground vehicles". In: *IEEE Internation Conference on Robotics and Automation*. May 2011, pp. 1732–1737. DOI: 10.1109/ICRA.2011.5979711.

[Low04] D. G. Lowe. "Distinctive image features from scale-invariant keypoints". In: *International Journal of Computer Vision* 60.2 (2004), pp. 91–110. URL: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.2.8899.

[Mar+08] Daniele Marzorati, Matteo Matteucci, Davide Migliore, and Domenico G. Sorrenti. "Monocular slam with inverse scaling parametrization". In: *Proceedings of 2008 British Machine Vision Conference (BMVC 2008)*. 2008, pp. 945–954.

[Mar+09] Daniele Marzorati, Matteo Matteucci, Davide Migliore, and Domenico G. Sorrenti. "On the use of inverse scaling in monocular slam". In: *IEEE Internation Conference on Robotics and Automation*. May 2009, pp. 2030–2036. DOI: 10.1109/ROBOT.2009.5152640.

[Mar11] Agostino Martinelli. "Vision and IMU Data Fusion: Closed-Form Solutions for Attitude, Speed, Absolute Scale and Bias Determination". Anglais. In: *IEEE Transactions on Robotics* (July 2011), Volume 28 (2012), Issue 1 (February), Issue 1 (February). URL: http://hal.archives-ouvertes.fr/hal-00743262.

[Mat+02] J. Matas, O. Chum, M. Urban, and T. Pajdla. “Robust wide baseline stereo from maximally stable extremal regions”. In: *Proceedings of BMVC02*. 2002, pp. 384–393. URL: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.7.2484.

[MCD06] J. M. M. Montiel, J. Civera, and A. J. Davison. “Unified inverse depth parametrization for monocular slam”. In: *Proceedings of Robotics: Science and Systems*. Philadelphia, USA, Aug. 2006.

[MD06] Jose M M Montiel and Andrew Davison. “A visual compass based on SLAM”. In: *IEEE Internation Conference on Robotics and Automation*. May 2006.

[Mei+11] Christopher Mei, Gabe Sibley, Mark Cummins, Paul Newman, and Ian Reid. “Rslam: a system for large-scale mapping in constant-time using stereo”. English. In: *International Journal of Computer Vision* 94 (2 2011), pp. 198–214. ISSN: 0920-5691. DOI: 10.1007/s11263-010-0361-7. URL: http://dx.doi.org/10.1007/s11263-010-0361-7.

[Mer04] Rudolph van der Merwe. “‘Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models’”. PhD thesis. Oregon Health & Science University, 2004.

[MFW04] J. Chris McGlone, Wolfgang Förstner, and Bernhard Wrobel. “‘Manual of Photogrammetry’”. In: asprs, 2004. Chap. 2 Mathematical Concepts in Photogrammetry.

[MR08] F. M. Mirzaei and S. I. Roumeliotis. “A kalman filter-based algorithm for imu-camera calibration: observability analysis and performance evaluation”. In: *IEEE Transaction on Robotics* 24(5) (Oct. 2008), pp. 1143–1156.

[MS04] K. Mikolajczyk and C. Schmid. “Scale and affine invariant interest point detectors”. In: *International Journal of Computer Vision* 60.1 (2004), pp. 63–86. DOI: 10.1023/B:VISI.0000027790.02288.f2. URL: http://portal.acm.org/citation.cfm?id=990402.

[Nüt+11] Gabriel Nützi, Stephan Weiss, Davide Scaramuzza, and Roland Siegwart. "Fusion of imu and vision for absolute scale estimation in monocular slam". In: *J. Intell. Robotics Syst.* 61.1-4 (Jan. 2011), pp. 287–299. ISSN: 0921-0296. DOI: 10.1007/s10846-010-9490-z. URL: http://dx.doi.org/10.1007/s10846-010-9490-z.

[Par62] Emanuel Parzen. "On estimation of a probability density function and mode". English. In: *The Annals of Mathematical Statistics* 33.3 (1962), pages. ISSN: 00034851. URL: http://www.jstor.org/stable/2237880.

[Paz+08] Lina M. Paz, Pedro Piniés, Juan D. Tardós, and José Neira. "Large-scale 6-dof slam with stereo-in-hand". In: *Transactions on Robotics* 24.5 (Oct. 2008).

[PB11] Arne Petersen and Marc-André Beyer. "Partitioned covariance intersection". In: *Proccedings of International Symposium Information on Ships*. Hamburg, Germany, Sept. 2011.

[PK10] Arne Petersen and Reinhard Koch. "Statistical analysis of kalman filters by conversion to gauss-helmert models with applications to process-noise estimation". In: *Proceedings of ICPR2010*. Istanbul, Turkey, Aug. 2010.

[PK12a] Arne Petersen and Reinhard Koch. "A novel state parametrization for stereo-slam". In: *Proceedings of the International Conference on Computer Vision Theory and Applications*. Vol. 2. Rome, Italy, Feb. 2012, pp. 144–153. ISBN: 978-989-8565-04-4.

[PK12b] Arne Petersen and Reinhard Koch. "Video-based realtime imu-camera calibration for robot navigation". In: ed. by Nasser Kehtarnavaz and Matthias F. Carlsohn. Vol. 8437. 1. Brussels, Belgium: SPIE, 2012, p. 843706. DOI: 10.1117/12.924066. URL: http://link.aip.org/link/?PSI/8437/843706/1.

[PWA06] Linthotage Dushantha Lochana Perera, Wijerupage Sardha Wijesoma, and Martin David Adams. "The estimation theoretic sensor bias correction problem in map aided localization". In: *The International Journal of Robotics Research* 25.7 (2006), pp. 645–667. DOI: 10.1177/0278364906066755.

[PWA10] L.D.L. Perera, W.S. Wijesoma, and M.D. Adams. “Slam with joint sensor bias estimation: closed form solutions on observability, error bounds and convergence rates”. In: *Control Systems Technology, IEEE Transactions on* 18.3 (May 2010), pp. 732–740. ISSN: 1063-6536. DOI: 10.1109/TCST.2009.2026165.

[Raw] RawSeeds. The RawSeeds Project. http://www.rawseeds.org, lastly visited November 2012.

[RS01] Vijay K Rohatgi and A K Saleh. An introduction to probability and statistics. 2nd ed. Wiley Series in Probability and Statistics. New York, NY: Wiley, 2001. ISBN: 0-471-34846-5.

[SBK08] Ingo Schiller, Christian Beder, and Reinhard Koch. “Calibration of a pmd camera using a planar calibration object together with a multi-camera setup”. In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. Vol. Vol. XXXVII. Part B3a. XXI. ISPRS Congress. Beijing, China, 2008, pp. 297–302.

[Sch] Ingo Schiller. MIP MultiCameraCalibration. www.mip.informatik.uni-kiel.de/tiki-index.php?page=Calibration; visited 1. nov. 2012.

[Sch+07] David Schleicher, Luis M. Bergasa, Rafael Barea, Elena López, Manuel Ocana, Jesús Nuevo, and Pablo Fernández. Real-Time Stereo Visual SLAM in Large-Scale Environments based on SIFT Fingerprints. Tech. rep. Department of Electronics, University of Alcalá, 2007.

[SF08] Richard Steffen and Wolfgang Förstner. “On visual real time mapping for unmanned aerial vehicles”. In: *IAPRSSIS* (2008), pp. 57–62.

[Sib+09] Gabe Sibley, Christopher Mei, Ian Reid, and Paul Newman. “Adaptive relative bundle adjustment”. In: *Robotics Science and Systems (RSS)*. Seattle, USA, June 2009.

[SLP07] N. Sünderhauf, S. Lange, and P. Protzel. "Using the unscented kalman filter in mono-slam with inverse depth parametrization for autonomous airship control". In: *Safety, Security and Rescue Robotics, 2007. SSRR 2007. IEEE International Workshop on*. Sept. 2007, pp. 1–6. DOI: 10.1109/SSRR.2007.4381265.

[SMD07] Joan Solá, Andrè Monin, and Michel Devy. "Bicamslam: two times mono is more than stereo". In: *Proceedings of IEEE International Conference on Robotics and Automation*. Rome, Italy, Apr. 2007, pp. 4795–4800.

[Sol+12] Joan Solà, Teresa Vidal-Calleja, Javier Civera, and José María Montiel. "Impact of landmark parametrization on monocular ekf-slam with points and lines". In: *Int. J. Comput. Vision* 97.3 (May 2012), pp. 339–368. ISSN: 0920-5691. DOI: 10.1007/s11263-011-0492-5. URL: http://dx.doi.org/10.1007/s11263-011-0492-5.

[Sor70] H. W. Sorenson. "Least-squares estimation: from gauss to kalman". In: *Spectrum, IEEE* 7.7 (July 1970), pp. 63–68. ISSN: 0018-9235. DOI: 10.1109/MSPEC.1970.5213471.

[SSM06] G. Sibley, G. Sukhatme, and L. Matthies. "The iterated sigma point kalman filter with applications to long range stereo". In: *Proceedings of Robotics: Science and Systems*. Philadelphia, USA, Aug. 2006.

[ST94] J. Shi and C. Tomasi. "Good features to track". In: *Conference on Computer Vision and Pattern Recognition*. IEEE. Seattle, June 1994, pp. 593–600. URL: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.36.2669.

[Sze11] Richard Szeliski. Computer Vision: Algorithms and Applications. Springer, 2011. ISBN: 978-1-84882-935-0.

[Ton90] Y. L. Tong. The Multivariate Normal Distribution. Springer, 1990. ISBN: 0-387-97062-2.

[Tul+08] Stephen Tully, Hyungpil Moon, George Kantor, and Howie Choset. "Iterated filters for bearing-only slam". In: *IEEE International Conference on Robotics and Automation*. 2008, pp. 1442–1448.

[TW04] David H. Titterton and John L. Weston. Strapdown Inertial Navigation Technology. 2nd ed. The Institution of Electrical Engineers, 2004. ISBN: 978-0-86341-358-2.

[Uhl03] Jeffrey K. Uhlmann. "Covariance consistency methods for fault-tolerant distributed data fusion". In: *Information Fusion* 4.3 (2003), pp. 201–215. ISSN: 1566-2535. DOI: 10.1016/S1566-2535(03)00036-8.

[WB06] Greg Welch and Gary Bishop. An Introduction to the Kalman Filter. Tech. rep. Chapel Hill, NC, USA: University of North Carolina at Chapel Hill, 2006.

[Wen07] Jan Wendel. Inertiale Navigationssysteme. Oldenburg Wissenschaftsverlag GmbH, 2007. ISBN: 978-3-486-58160-7.

[Wil+09] Brian Williams, Mark Cummins, José Neira, Paul Newman, Ian Reid, and Juan Tardós. "A comparison of loop closing techniques in monocular slam". In: *Robot. Auton. Syst.* 57.12 (Dec. 2009), pp. 1188–1197. ISSN: 0921-8890. DOI: 10.1016/j.robot.2009.06.010.

[Wu07] Changchang Wu. SiftGPU: A GPU Implementation of Scale Invariant Feature Transform (SIFT). http://cs.unc.edu/~ccwu/siftgpu. 2007.

[YM09] Guoshen Yu and J.-M. Morel. "A fully affine invariant image comparison method". In: *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on.* Apr. 2009, pp. 1597–1600. DOI: 10.1109/ICASSP.2009.4959904.