Streamlining Calibration: Eliminating Infrastructure and Certifying Optimality

by

Emmett S. Wise

A thesis submitted in conformity with the requirements
for the degree of Doctor of Philosophy:
Graduate Department of Institute for Aerospace Studies
University of Toronto

Abstract

Streamlining Calibration: Eliminating Infrastructure and Certifying Optimality

Emmett S. Wise
Doctor of Philosophy
Graduate Department of Institute for Aerospace Studies
University of Toronto
2025

Autonomous systems often fuse data from multiple sensors and sensing modalities to improve robustness when operating in adverse conditions. Correct sensor fusion requires knowledge of the transformations between the sensor reference frames as well as the temporal offsets between the sensor measurement times. Due to modifications and wear-and-tear, estimates of these parameters may become inaccurate over time, and end-users of autonomous systems in turn require processes to estimate these parameters. The process of estimating the spatial transformation is known as extrinsic calibration, while the process of estimating the temporal offset is known as temporal calibration. Jointly estimating both sets of parameters is known as spatiotemporal calibration. Many state-of-the-art calibration methods require specialized targets and rough initial guesses for the parameters of interest. These requirements limit potential sensor configurations and calibration venues (e.g., the environment must contain a target and all sensors must view that singular target). In this thesis, we seek to lift these restrictions and streamline the calibration process for end-users. Initially, we explore a method to eliminate specialized targets in spatiotemporal calibration where one sensor is a radar. We then develop a targetless extrinsic calibration method for pairs of radars. Additionally, we apply recent results from convex optimization to two classic calibration problems, the hand-eye calibration problem and the hand-eye-robot-world calibration problem, yielding solutions for the optimal calibration parameters (for a dataset) without prior knowledge. Through simulation studies and real-world experiments, we demonstrate that our methods achieve estimation accuracy similar to or better than other calibration methods that require initialization or specialized targets.

Acknowledgements

This document is the culmination of many years of support from family, friends, and mentors. First and foremost, I would like to express my gratitude to my parents, Penny and Gord, for their unwavering support and faith in me during this lengthy journey. I am also thankful for the encouragement provided by my sister, Marleigh, and aunt, Tina. Hannah, your support during the late-night paper submissions, long TA hours, and exam study sessions was essential in helping me complete this thesis. I am grateful to my friends for providing a social environment where I could momentarily set aside the pressures of graduate school. I want to thank my colleagues in STARS laboratory for providing a productive and fun working environment. In particular, the suggestions from Matthew Giamou and Juraj Peršić were critical in helping me find my footing in robotics research.

During my academic career, I have been lucky to have many mentors. By fostering my love of math and robotics, my work has grown from constructing line-following robots that fit in shoe boxes to the research presented in this thesis. I want to thank my committee members Tim Barfoot and Steve Waslander for pushing me to produce the highest quality research. Finally, this thesis would not have been possible without the guidance and encouragement provided by Jonathan.

Contents

Notation

a	:	Symbols in this font are real scalars.
$\mathbf{a}$	:	Symbols in this font are real column vectors.
$\dot{\mathbf{a}}, \ddot{\mathbf{a}}, \mathbf{a}^{(i)}$	:	the first, second and ith time-derivative of $\mathbf{a}$.
$\mathbf{A}$	:	Symbols in this font are real matrices.
$\mathbf{A}_{(i,j)}$	:	The ith row and the jth column of the matrix $\mathbf{A}$.
$\mathbf{A}_{(i:j,k:l)}$	:	A submatrix of $\mathbf{A}$ that contains the entries from the $[i, j]$ rows and $[k, l]$ columns.
$\mathbf{A}_{(i,:)}$	:	The ith row of the matrix $\mathbf{A}$.
$\mathbf{A}_{(:,j)}$	:	The jth column of the matrix $\mathbf{A}$.
$\mathcal{N}(\boldsymbol{\mu}, \mathbf{R})$	:	Normally distributed with mean $\boldsymbol{\mu}$ and covariance $\mathbf{R}$.
$\text{Lang}(\mathbf{A}, \kappa)$	:	Langevin distributed with mode $\mathbf{A}$ and concentration parameter κ.
$\underrightarrow{\boldsymbol{\mathcal{F}}}_a$	:	A reference frame in three dimensions.
$\otimes$	:	The kronecker product.
$\times$	:	The cross product.
$(\cdot)^\wedge$	:	This operator acts on $\mathbf{r} \in \mathbb{R}^3$ to produce a skew-symmetric matrix such that $\mathbf{r}^\wedge\mathbf{s}$ is equivalent to the cross product $\mathbf{r} \times \mathbf{s}$.
$(\cdot)^\vee$	:	The inverse operation of $(\cdot)^\wedge$.
$\mathbf{I}_n$	:	The n by n identity matrix.
$\mathbf{0}$	:	The zero matrix.
$\mathbf{p}_a^{ba}$	:	The Vector from a to b expressed in $\underrightarrow{\boldsymbol{\mathcal{F}}}_a$.
$\mathbf{R}_{ab}$	:	The rotation matrix that rotates vectors from $\underrightarrow{\boldsymbol{\mathcal{F}}}_b$ to $\underrightarrow{\boldsymbol{\mathcal{F}}}_a$.
$\mathbf{T}_{ab}$	:	The 4 by 4 transformation matrix that transforms homogeneous points from $\underrightarrow{\boldsymbol{\mathcal{F}}}_b$ to $\underrightarrow{\boldsymbol{\mathcal{F}}}_a$.
$\boldsymbol{\omega}_a^{ba}$	:	The rotational velocity of frame $\underrightarrow{\boldsymbol{\mathcal{F}}}_b$ relative to $\underrightarrow{\boldsymbol{\mathcal{F}}}_a$, expressed in $\underrightarrow{\boldsymbol{\mathcal{F}}}_a$.
$\nabla_{\mathbf{x}}$	:	The operator that returns the gradient with respect to $\mathbf{x}$.
$\exp(\cdot)$	:	The operator that maps the Lie Algebra to the Lie Group.
$\log(\cdot)$	:	The inverse operator of $\exp(\cdot)$.
$\text{vec}\,(\cdot)$	:	The function that unwraps matrix in column-major order
$\text{cyclic}\,(\mathbf{s})$	:	The space of cyclic permutations of the set $\mathbf{s}$.
$\text{tr}\,(\cdot)$	:	The operator that sums the diagonal of a matrix.
$\text{blkdiag}\,(\cdot)$	:	The operator that constructs a block diagonal matrix.
S^n	:	The unit n-sphere, which is $\{\mathbf{x} \in \mathbb{R}^{n+1} \mid \|\mathbf{x}\|_2 = 1\}$.
$\mathbb{S}^n$	:	The space of $n \times n$ real symmetric matrices.
$\mathrm{SO}(3)$	:	The special orthogonal group, which we use to represent rotations.
$\mathrm{SE}(3)$	:	The special euclidean group, which we use to represent poses.
$\mathbf{A} \succeq 0$	:	The curled inequality and the strict equivalent $\succ$ indicate that the matrix is either positive semidefinite or definite, respectively.

Chapter 1

Introduction

Successful, widespread deployment of autonomous vehicles (AVs) depends on their ability to operate safely under a range of challenging environmental conditions. Poor operating conditions can corrupt the measurements of an AV and may mask crucial elements of a given scene, such as pedestrians walking front of a vehicle. To ensure sufficient redundancy, most AV perception systems fuse data from multiple sensors and sensing modalities and are therefore more complex than solutions that rely on an individual sensor. Such sensor fusion requires knowledge not only of the spatial transformations among sensor reference frames but also of the temporal offsets among their times. While manufacturers can measure the spatial transformations and minimize the temporal offsets through careful engineering, these parameters may change over time due to end-user modification or wear and tear.

Since returning robots to the factory for adjustments is both expensive and time-consuming, end-users require sensor calibration techniques that can be carried out both online and in the field. The process of determining the spatial transformation between two sensor reference frames is typically referred to as *extrinsic calibration*. While multiple sensors are observing a scene, if inaccurate spatial transformations are applied to express the data in one common reference frame, then portions of the scene will appear spatially shifted in different sensor datastreams. In addition to extrinsic calibration, users require a process to temporally align sensor datastreams, which is usually known as *temporal calibration*. Even when the sensors are externally timed and triggered, internal signal processing delays may result in shifted measurement timestamps. For some systems, power cycling or reconfiguring the sensors can also change these time offsets. When in motion, incorrect temporal offsets may lead to *spatial* shifts across sensor datastreams, as is the case with inaccurate spatial transformations. Finally, *spatiotemporal calibration* is the process of estimating the spatial transformation(s) and temporal offset(s) together.

Because sensor calibration is essential to robot autonomy, problems related to it have been studied extensively. Many methods to address it, however, have a variety of environmental or system requirements that restrict viable calibration scenarios. For example, most rely on the alignment of observed, identifiable features across sensor pairs, which requires the sensors to have overlapping fields of view and for the environment to contain specialized targets, thereby limiting potential sensor configurations and calibration venues. Further, most calibration processes find a set of parameters that minimize erroneous spatial shifts when expressing sensor data in different reference frames. However, a calibration process may return a local minimum. To avoid local minima, these processes require a skilled human operator to provide initial estimates of the spatial transformations and temporal offsets, which must be close to the ground truth values. Consequently, autonomous

systems are often unable to perform calibration in-situ and end-users must transport their robot to a central location for maintenance.

In this thesis, we improve upon the state of the art in robotic calibration by removing configuration restrictions, relaxing calibration venue requirements, and developing calibration processes that do not require human-aided initialization. We focus a portion of our work on calibration involving mm-wavelength radar sensors, which are useful for a variety of robotics applications. Additionally, we develop methods to solve two classic calibration problems with no initialization requirements. Below, we present a brief overview of the calibration literature related to the research presented in this thesis, summarize the thesis structure, and discuss our novel contributions.

1.1 Radar Calibration

Radar sensors operate by actively emitting electromagnetic radiation and measuring its reflection from surfaces in the environment. These sensors are relatively immune to the adverse conditions that negatively affect more commonly used sensors, such as cameras. Further, radar sensors provide information that cameras cannot, including *range-rate* measurements to targets in the environment. However, radar data are much lower resolution and have significantly more noise, than visual measurements under nominal conditions. Additionally, many radars have narrow fields of view, so multiple radars are often required for complete coverage of the surrounding environment. Together, radars and cameras are highly complementary, providing situational awareness under nominal and visually degraded conditions. Consequently, the calibration of radar-camera and radar pairs is an important area of research.

Generally, calibration algorithms for radar-camera and radar pairs estimate calibration parameters by minimizing the misalignment between mutually observed targets or landmarks in the surrounding environment. Reprojection-based methods assume that the radar measures point-like reflections from objects. As shown in Richards et al. [2010], a radar measurement (of the reflected electromagnetic pulse) is a complex function of the shape, relative orientation, size, and composition of an object. As a result, the measurements are not in general point-like and issues such as ghosting (due to multipath reflections) can complicate the use of their data. To avoid such problems, we use specialized trihedral retroreflective targets (see Figure 1.1) to ensure point-like radar returns. Often, the specialized targets also have a visual fiducial, which simplifies the radar-camera measurement correspondence problem. The use of targets, however, means that calibration must be carried out in specialized areas or with infrastructure that is not usually available during regular AV operation. Further, the sensor pairs must share overlapping fields of view, which may not be possible for all system configurations.

To eliminate the target requirement and the need for cross-modal data association, a subset of radar calibration algorithms rely on the range-rate measurements of the radar. In radar data, the range-rate measurement is the radial velocity of a target relative to the reference frame of the radar. Using the rigid-body kinematic equations, Kellner et al. [2013] and Doer and Trommer [2020b] relate the range-rate measurements with data from another sensor, which enables targetless calibration of the sensor pair. We use this approach to develop extrinsic and spatiotemporal calibration algorithms for radar-camera and radar pairs. Further, we prove that the radar-camera and radar-radar systems are *observable*, which is a necessary property for reliable calibration.

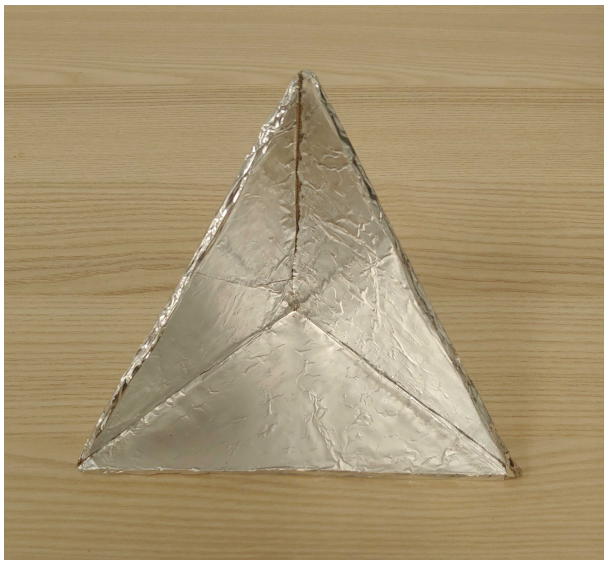


Figure 1.1: A radar trihedral corner reflector. The corner formed by the three metal panels ensures a strong radar reflection that is readily detected. This target simplifies radar and cross-modal data association.

1.2 Convex Relaxations and Certifiable Calibration

To estimate the calibration parameters of a robotic system, many calibration algorithms combine the data and then solve a nonlinear, nonconvex optimization problem. Given a set of data, the solvers return the 'best' parameters that are typically close to a user-provided initial guess. In some cases, we can develop heuristic initialization schemes to provide a reasonable initial guess for some or all of the parameter values. If initialization fails (i.e., the initial estimates are poor), then the estimated parameters can be a local minima and deviate from reality. Ideally, to enable complete autonomy, systems must be able to calibrate without human intervention. Consequently, one area of research is *certifiable calibration* or algorithms that can determine or *certify* if the estimated parameters are the global minimum for a given dataset.

To enable certifiable calibration, we leverage semidefinite program (SDP) relaxations, in which we relax nonconvex calibration problems into convex optimization problems. For convex optimization problems, any local minimum is the global minimum, so we always find the 'best' solution and do not require an accurate initial guess. In some cases, the global minimum of the convex relaxation is also the global minimum of the nonconvex calibration problem. Consequently, we can determine the solution to some nonconvex calibration problems by solving the convex relaxation, which does not require accurate initialization. Convex SDP relaxations have become a popular solution method for a number of geometric estimation problems including rotation averaging (e.g., Yang et al. [2021]), SLAM (e.g., Rosen et al. [2019], Briales and Gonzalez-Jimenez [2017a]), registration (e.g., Briales and Gonzalez-Jimenez [2017b]), relative pose estimation (e.g., Briales et al. [2018]), and calibration (e.g., Heller et al. [2014], Giamou et al. [2019], Wodtko et al. [2021], Horn et al. [2023]). In this thesis, we develop certifiable calibration methods for monocular cameras mounted on mobile robots.

1.3 Thesis Structure and Contributions

The thesis is structured as follows. Chapters 2 to 4 review the mathematical machinery required to understand the remaining chapters. In particular, Chapter 2 reviews key mathematical concepts and Chapter 3 presents

several ways to formulate and solve various calibration problems. In Chapter 4, we define observability and identifiability.

In Chapters 5 and 6, we present two approaches for calibrating sensor pairs where at least one sensor is a radar. Chapter 5 presents a spatiotemporal calibration method for 3D mm-wavelength radar-camera pairs and is based on the two publications: Wise et al. [2021] and Wise et al. [2023]. Chapter 6 discusses an extrinsic calibration method for pairs of 2D mm-wavelength radars and is based on the publication Cheng et al. [2023].

In Chapters 7 and 8, we present certifiable methods for two calibration problems. Chapter 7 presents a certifiably optimal approach for the extrinsic calibration of a monocular camera mounted on a mobile robot and is based on Wise et al. [2020]. Chapter 8 presents a certifiably optimal approach for extrinsic calibration of many monocular cameras mounted on a mobile robot or fixed in an environment. The calibration approach in Chapter 7 requires camera motion, which is not possible in fixed camera calibration. Consequently, the many-camera calibration method subsumes the single-camera method. This work is not yet submitted for publication.

Finally, Chapter 9 summarizes the contributions of this thesis and discusses several potential future research directions.

Chapter 2

Foundations

In this chapter, we review concepts and definitions relevant to the findings in this thesis. Section 2.1 discusses some mathematical preliminaries, while Section 2.2 defines a number of important probability distributions. In Section 2.3, we provide a short taxonomy of types of calibration problems. Section 2.4 describes calibration methods based on homography, reprojection, and motion. Finally, in Section 2.5 we consider several abstracted sensor models for motion-based calibration, which summarize raw measurements into a representation that is suitable for specific calibration tasks.

2.1 Mathematical Preliminaries

Several operators and geometrical concepts are required for this study, including coordinate frames, rigid-body transformations, and a B-spline trajectory representation.

2.1.1 Operator Definitions

We use a number of operators to simplify expressions in the chapters that follow. The wedge operator $(\cdot)^\wedge$ is defined as

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix}^\wedge = \begin{bmatrix} 0 & -z & y \\ z & 0 & -x \\ -y & x & 0 \end{bmatrix}, \tag{2.1}$$

such that the cross product of two vectors can be expressed by

$$\mathbf{a} \times \mathbf{b} = \mathbf{a}^\wedge \mathbf{b}. \tag{2.2}$$

The vee operator $(\cdot)^\vee$ is defined as the inverse of the wedge operator,

$$\begin{bmatrix} 0 & -z & y \\ z & 0 & -x \\ -y & x & 0 \end{bmatrix}^\vee = \begin{bmatrix} x \\ y \\ z \end{bmatrix}. \tag{2.3}$$

Let the 'circle times' operator $\mathbf{A} \otimes \mathbf{B}$ define the Kronecker product of the matrices $\mathbf{A}$ and $\mathbf{B}$. If $\mathbf{A}$ and $\mathbf{B}$ are p by m and q by n, respectively, then

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} a_{11}\mathbf{B} & \cdots & a_{1m}\mathbf{B} \\ \vdots & \ddots & \vdots \\ a_{p1}\mathbf{B} & \cdots & a_{pm}\mathbf{B} \end{bmatrix}. \tag{2.4}$$

One particularly useful Kronecker product identity is

$$\text{vec}\left(\mathbf{AXB}\right) = \left(\mathbf{B}^{\mathsf{T}} \otimes \mathbf{A}\right) \text{vec}\left(\mathbf{X}\right). \tag{2.5}$$

We use $\oplus$ to represent the Kronecker sum

$$\mathbf{A} \oplus \mathbf{B} = \mathbf{A} \otimes \mathbf{I} + \mathbf{I} \otimes \mathbf{B}, \tag{2.6}$$

where $\mathbf{A} \oplus \mathbf{B} \in \mathbb{R}^{pq \times mn}$.

2.1.2 Translations, Rotations, and Rigid-Body Transformations

Let $\underrightarrow{\mathcal{F}}_a$ and $\underrightarrow{\mathcal{F}}_b$ be two d-dimensional reference frames, where $d = 2$ or 3. Following the notation in Barfoot [2024], we denote the translation vector from $\underrightarrow{\mathcal{F}}_a$ to $\underrightarrow{\mathcal{F}}_b$, expressed in $\underrightarrow{\mathcal{F}}_a$, as $\mathbf{t}_a^{ba} \in \mathbb{R}^d$. Figure 2.1 shows a diagram of the translation from $\underrightarrow{\mathcal{F}}_a$ to $\underrightarrow{\mathcal{F}}_b$, where $\underrightarrow{\mathcal{F}}_a$ is stationary and $\underrightarrow{\mathcal{F}}_b$ is attached to a vehicle. If $\underrightarrow{\mathcal{F}}_b$ and $\underrightarrow{\mathcal{F}}_a$ are rotated relative one another, then the corresponding rotation matrix from $\underrightarrow{\mathcal{F}}_b$ to $\underrightarrow{\mathcal{F}}_a$ is $\mathbf{R}_{ab}$.

Rotation matrices are elements of the special orthogonal group:

$$\text{SO}(d) : \mathbf{R}_{ab} \in \mathbb{R}^{d \times d}, \tag{2.7}$$

$$\mathbf{R}_{ab}{}^{\mathsf{T}}\mathbf{R}_{ab} = \mathbf{R}_{ab}\mathbf{R}_{ab}{}^{\mathsf{T}} = \mathbf{I}_{d \times d}, \tag{2.8}$$

$$\det(\mathbf{R}_{ab}) = 1. \tag{2.9}$$

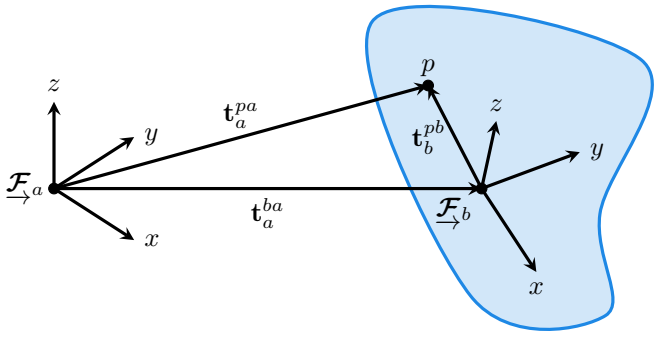


Figure 2.1: Illustration of the relationship between two 3D coordinate frames $\underrightarrow{\mathcal{F}}_a$ and $\underrightarrow{\mathcal{F}}_b$, where $\underrightarrow{\mathcal{F}}_b$ is attached to a vehicle. The translation and rotation between the frames are $\mathbf{t}_a^{ba}$ and $\mathbf{R}_{ab}$, respectively. Further, the translation vectors to point p from the origin of $\underrightarrow{\mathcal{F}}_a$ and $\underrightarrow{\mathcal{F}}_b$ are $\mathbf{t}_a^{pa}$ and $\mathbf{t}_b^{pb}$, respectively.

When $d = 3$, the constraint $\det(\mathbf{R}_{ab}) = 1$ can be replaced by

$$\mathbf{R}_{ab,(:,i)} \times \mathbf{R}_{ab,(:,j)} = \mathbf{R}_{ab,(:,k)}, \; i,j,k \in \text{cyclic}(1,2,3). \tag{2.10}$$

Elements of $\mathrm{SO}(d)$ satisfy the group axioms,

$$\begin{aligned}
\text{Closure:} \quad & \mathbf{R}_1\mathbf{R}_2 \in \mathrm{SO}(d) \; \forall \; \mathbf{R}_1, \mathbf{R}_2 \in \mathrm{SO}(d), \\
\text{Associativity:} \quad & \mathbf{R}_1 \left(\mathbf{R}_2\mathbf{R}_3\right) = \left(\mathbf{R}_1\mathbf{R}_2\right)\mathbf{R}_3 \; \forall \; \mathbf{R}_1, \mathbf{R}_2, \mathbf{R}_3 \in \mathrm{SO}(d), \\
\text{Identity:} \quad & \mathbf{R}\mathbf{I} = \mathbf{I}\mathbf{R} \; \forall \; \mathbf{R} \in \mathrm{SO}(d), \\
\text{Invertibility:} \quad & \mathbf{R}^{-1} \in \mathrm{SO}(d) \; \forall \; \mathbf{R} \in \mathrm{SO}(d).
\end{aligned}$$

Notably, these group operations are smooth, and so the special orthogonal group $\mathrm{SO}(d)$ is a Lie group.

When elements of $\mathrm{SO}(d)$ are expressed as matrices, the constraints on the matrix entires can complicate certain optimization problems. Thus, we will sometimes use non-unique lower-dimensional representations of rotations. In this thesis, our models contain elements of $\mathrm{SO}(3)$ or $\mathrm{SO}(2)$, so we begin by defining a lower-dimensional representation of $\mathrm{SO}(3)$. Since all Lie groups have an associated Lie algebra, we can parameterize elements of $\mathrm{SO}(3)$ close to the identity element through the Lie algebra of the group,

$$\mathfrak{so}(3) = \phi\,\mathbf{a}^{\wedge}, \tag{2.11}$$

where $\phi \in \mathbb{R}$ and $\mathbf{a} \in \mathrm{S}^2$. From Barfoot [2024], the mapping $\exp : \mathfrak{so}(3) \to \mathrm{SO}(3)$ is defined by

$$\mathbf{R} = \exp\left(\phi\,\mathbf{a}^{\wedge}\right) = \cos\phi\,\mathbf{I} + (1 - \cos\phi)\,\mathbf{a}\mathbf{a}^{\mathsf{T}} + \sin\phi\,\mathbf{a}^{\wedge}. \tag{2.12}$$

The vector space representation of (small) rotations as $\phi\,\mathbf{a}$ is also known as the angle-axis representation. When two coordinate frames are constrained to lie in the same plane of rotation (i.e., they share one common axis), the 2D rotation between the frames is an element of $\mathrm{SO}(2)$. We can parameterize $\mathrm{SO}(2)$ using its Lie algebra,

$$\mathfrak{so}(2) = \phi \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}. \tag{2.13}$$

The mapping $\exp : \mathfrak{so}(2) \to \mathrm{SO}(2)$ is

$$\exp\left(\phi^{\wedge}\right) = \begin{bmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{bmatrix}. \tag{2.14}$$

The inverse mapping from $\mathrm{SO}(2) \to \mathfrak{so}(2)$ is trivial, but the inverse (logarithmic) map $\log : \mathrm{SO}(3) \to \mathfrak{so}(3)$ is not as straightforward. Using the method in Barfoot [2024], we start by determining the axis of rotation. Since $\mathbf{R}\mathbf{a} = \mathbf{a}$, the axis of rotation is the unit eigenvector of $\mathbf{R}$ with an eigenvalue of 1. Enforcing the convention that $|\,\phi\,| < \pi$, the angle for a given rotation matrix is

$$\phi = \cos^{-1}\left(\frac{\mathrm{tr}\,(\mathbf{R}) - 1}{2}\right). \tag{2.15}$$

For brevity, we use $\boldsymbol{\phi}^{\wedge}$ to represent elements of $\mathfrak{so}(3)$.

When the transcendental functions in the mapping from $\mathfrak{so}(3)$ to $\mathrm{SO}(3)$ complicate mathematical analyses,

we leverage the unit quaternion parameterization. A unit quaternion is a hypercomplex number,

$$\mathbf{q}_{ab} = q_0 + q_1 i + q_2 j + q_3 k \tag{2.16}$$

for which $i^2 = j^2 = k^2 = ijk = -1$ and $q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1$. Similar to Solà et al. [2021], we use the 'vector' representation of the unit quaternion,

$$\mathrm{S}^3 : \mathbf{q}_{ab} \in \mathbb{R}^4, \tag{2.17}$$

$$\mathbf{q}_{ab}{}^{\mathsf{T}} \mathbf{q}_{ab} = 1. \tag{2.18}$$

Specifically, we split the unit quaternion into real, q_0, and complex, $\mathbf{q}_v{}^{\mathsf{T}} = \begin{bmatrix} q_1 & q_2 & q_3 \end{bmatrix}^{\mathsf{T}}$ components, where

$$\mathbf{q}_{ab}{}^{\mathsf{T}} = \begin{bmatrix} q_0 & \mathbf{q}_v{}^{\mathsf{T}} \end{bmatrix}^{\mathsf{T}}. \tag{2.19}$$

While we do not make use of it, it is interesting to note that the group S^3 is also a Lie group: readers looking for more information should consult Solà [2017] or Solà et al. [2021]. The mapping $\mathrm{S}^3 \to \mathrm{SO}(3)$ is

$$\mathbf{R}\left(\mathbf{q}_{ab}\right) = \left(2q_0^2 - 1\right)\mathbf{I}_3 + 2\mathbf{q}_v\mathbf{q}_v{}^{\mathsf{T}} + 2q_0\mathbf{q}_v{}^{\wedge}. \tag{2.20}$$

Notably, the manifold of unit quaternions doubly covers the manifold of rotation matrices (i.e., $\mathbf{R}(\mathbf{q}) = \mathbf{R}(-\mathbf{q})$), but the antipodal solutions do not affect our mathematical analyses.

Finally, we construct the rigid-body transformation $\mathbf{T}_{ab}$ from rotation and translation components,

$$\mathbf{T}_{ab} = \begin{bmatrix} \mathbf{R}_{ab} & \mathbf{t}_a^{ba} \\ \mathbf{0} & 1 \end{bmatrix}. \tag{2.21}$$

2.1.3 Rotation Jacobians

In Chapter 5, solving the spatiotemporal radar-camera calibration problem requires the Jacobians of various functions with respect to rotation parameters. This subsection explores two approaches for determining the Jacobian of a (rotated) vector with respect to the rotation matrix itself, namely

$$\frac{\partial \mathbf{R}\mathbf{p}}{\partial \boldsymbol{\phi}}, \tag{2.22}$$

where $\mathbf{R} = \exp\left(\boldsymbol{\phi}^{\wedge}\right) \in \mathrm{SO}(3)$ and $\mathbf{p} \in \mathbb{R}^3$. In the first approach, the Jacobian is evaluated with respect to a non-unique lower dimensional parameterization. As shown in Barfoot [2024], when we parameterize a rotation using $\mathfrak{so}(3)$, the Jacobian of a rotated vector with respect to the Lie algebra is

$$\frac{\partial \mathbf{R}\mathbf{p}}{\partial \boldsymbol{\phi}} = -(\mathbf{R}\mathbf{p})^{\wedge}\mathbf{J}_l, \tag{2.23}$$

where $\mathbf{J}_l$ is the left Jacobian of $\mathbf{R}$,

$$\mathbf{J}_l = \frac{\sin\phi}{\phi}\mathbf{I} + \left(1 - \frac{\sin\phi}{\phi}\right)\mathbf{a}\mathbf{a}^{\mathsf{T}} + \frac{1 - \cos\phi}{\phi}\mathbf{a}^{\wedge}. \tag{2.24}$$

We can also define the right Jacobian of $\mathbf{R}$,

$$\mathbf{J}_r = \frac{\sin\phi}{\phi}\mathbf{I} + \left(1 - \frac{\sin\phi}{\phi}\right)\mathbf{a}\mathbf{a}^\intercal - \frac{1-\cos\phi}{\phi}\mathbf{a}^\wedge. \tag{2.25}$$

From Solà [2017], when we parameterize $\mathrm{SO}(3)$ with unit quaternions, the Jacobian of a (rotated) vector with respect to the unit quaternion is

$$\frac{\partial \mathbf{R}(\mathbf{q})\mathbf{p}}{\partial \mathbf{q}} = \begin{bmatrix} (4q_0\mathbf{I}_3 + 2\mathbf{q}_v{}^\wedge)\mathbf{p} & 2((\mathbf{q}_v{}^\intercal\mathbf{p})\mathbf{I}_3 + \mathbf{q}_v\mathbf{p}^\intercal - q_0\mathbf{p}^\wedge) \end{bmatrix}. \tag{2.26}$$

Equations (2.23) and (2.26) are used in our analyses of systems in Chapter 5.

For the optimization of problems described in Chapter 5, we compute the Jacobian of a rotation matrix with respect to a perturbation in the Lie algebra. For a nominal rotation, $\boldsymbol{\phi}$, and Lie algebra perturbation, $\boldsymbol{\psi}$, the perturbation can be applied in three different ways,

$$\begin{aligned} &\exp\left(\boldsymbol{\psi}^\wedge\right)\exp\left(\boldsymbol{\phi}^\wedge\right) && \text{(left perturbed)},\\ &\exp\left((\boldsymbol{\phi}+\boldsymbol{\psi})^\wedge\right) && \text{(middle perturbed)},\\ &\exp\left(\boldsymbol{\phi}^\wedge\right)\exp\left(\boldsymbol{\psi}^\wedge\right) && \text{(right perturbed)}. \end{aligned}$$

While this method extends to quaternions, we do not make use of the quaternion perturbation approach. Following the convention in Barfoot [2024], we choose to compute the derivatives with respect to the left perturbation,

$$\frac{\partial \exp\left(\boldsymbol{\psi}^\wedge\right)\mathbf{R}\mathbf{p}}{\partial \boldsymbol{\psi}} = -(\mathbf{R}\mathbf{p})^\wedge. \tag{2.27}$$

Equation (2.27) does not require the left Jacobian, $\mathbf{J}_l$, which simplifies our optimization routines. Conveniently, the perturbation scheme extends to the directional derivatives of functions involving rotation matrices, so this perturbation approach is used in many optimization problems.

2.1.4 Rotational Kinematics

Radar calibration methods that leverage range-rate information require a kinematic model of $\mathrm{SO}(3)$. The time derivative of a rotation matrix is

$$\dot{\mathbf{R}}_{ab} = \mathbf{R}_{ab}\boldsymbol{\omega}_b^{ba}{}^\wedge = -\boldsymbol{\omega}_a^{ab}{}^\wedge\mathbf{R}_{ab}, \tag{2.28}$$

where $\boldsymbol{\omega}_b^{ba}$ is the rotational velocity of $\underrightarrow{\mathcal{F}}_b$ relative to $\underrightarrow{\mathcal{F}}_a$ expressed in $\underrightarrow{\mathcal{F}}_b$. The relationship between the rotational velocity and the time derivative of the Lie algebra is

$$\boldsymbol{\omega}_a^{ab} = -\mathbf{J}_{ab}\dot{\boldsymbol{\phi}}_{ab}, \tag{2.29}$$

where $\mathbf{J}_{ab}$ is the left Jacobian of $\mathbf{R}_{ab}$. Finally, using the definitions in Solà [2017], the time derivative of the unit quaternion $\mathbf{q}_{ab}$ is

$$\dot{\mathbf{q}}_{ab} = \frac{1}{2}\boldsymbol{\Xi}(\mathbf{q}_{ab})\,\boldsymbol{\omega}_b^{ba} = \frac{1}{2}\boldsymbol{\Omega}(\boldsymbol{\omega}_b^{ba})\,\mathbf{q}_{ab}, \tag{2.30}$$

$$\boldsymbol{\Omega}(\boldsymbol{\omega}_b^{ba}) = \begin{bmatrix} 0 & -\boldsymbol{\omega}_b^{ba\intercal} \\ \boldsymbol{\omega}_b^{ba} & -\boldsymbol{\omega}_b^{ba}{}^\wedge \end{bmatrix}, \tag{2.31}$$

$$\boldsymbol{\Xi}(\mathbf{q}_{ab}) = \begin{bmatrix} -\mathbf{q}_v^{\mathsf{T}} \\ q_0 \mathbf{I}_3 + \mathbf{q}_v^{\wedge} \end{bmatrix}. \tag{2.32}$$

2.1.5 B-Splines for Continuous-Time Trajectories

In Chapter 5, we query the rigid-body pose of a sensor platform at arbitrary points in time. To enable this, we parameterize the time-varying trajectory of the platform using the B-spline representation from Sommer et al. [2020]. We briefly review B-splines below and refer the reader to Sommer et al. [2020], de Boor [1978], and Qin [1998] for additional details.

A B-spline of order k is a function of one continuous parameter (e.g., time) and a finite set of control points; for brevity, we restrict this example here to control points $\{\mathbf{p}_0, \ldots, \mathbf{p}_N \mid \mathbf{p}_i \in \mathbb{R}^d\}$. In a uniformly-spaced B-spline, each control point is assigned a time (or *knot*) $t_i = t_0 + i\Delta t$, where t_0 marks the beginning of the spline and Δt is the time between knots. Evaluating a k^{th} order B-spline at time t, where $t_i \leq t < t_{i+1}$, requires the set of k control points over the knot sequence $t_i, \ldots, t_{i+k-1}$. As a result, the end point of a B-spline of length N and order k is at time t_{N-k+1}. An example of an $\mathbb{R}^3$ B-spline of order 4 is shown in Figure 2.2.

The first step in computing the value of a k^{th} order B-spline at time t is to convert t to the normalized time $u = \frac{t-t_i}{t_{i+1}-t_i}$. Given u, the value of the k^{th} order B-spline is defined as

$$\mathbf{p}(u) = \begin{bmatrix} \mathbf{p}_i & \mathbf{d}_1^i & \ldots & \mathbf{d}_{k-1}^i \end{bmatrix} \tilde{\mathbf{M}}_k \mathbf{u}, \tag{2.33}$$

where $\mathbf{u}^T = [1 \; u \; u^2 \; \ldots \; u^{k-1}]$ and $\mathbf{d}_j^i = \mathbf{p}_{i+j} - \mathbf{p}_{i+j-1}$. The elements of the $k \times k$ mixing matrix, $\tilde{\mathbf{M}}_k$, are

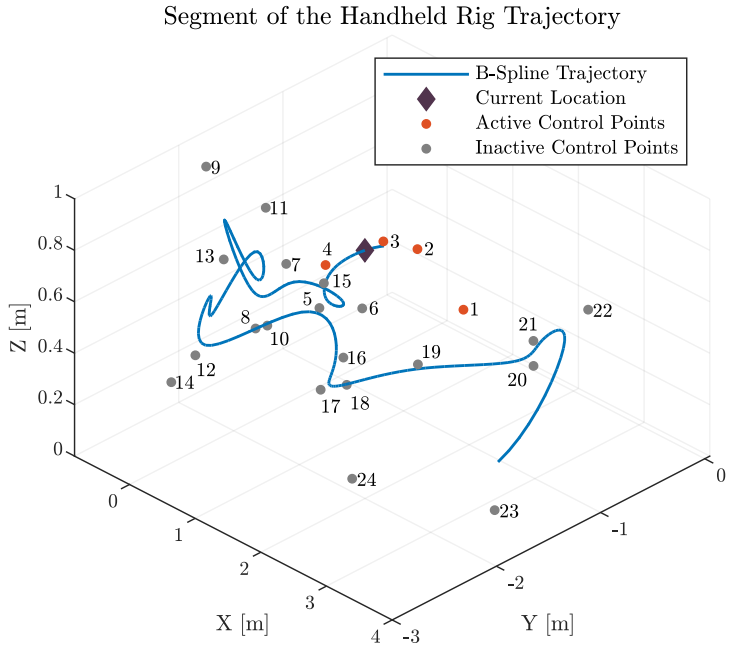


Figure 2.2: A segment of an estimated $\mathbf{r}_r^{wr}$ (position) B-spline, where $\underrightarrow{\mathcal{F}}_r$ and $\underrightarrow{\mathcal{F}}_w$ are radar and stationary world reference frames, respectively. This particular B-spline was estimated from radar and camera data collected using the handheld rig in Section 5.6.2. The purple diamond is the position of the rig 6.3 s from the start of the trajectory. The active control points at 6.3 s are shown in orange. As the rig continues along the trajectory, the active nodes change.

defined by

$$\tilde{\mathbf{M}}_k^{(a,n)} = \sum_{s=a}^{k-1} m_k^{(s,n)}, \tag{2.34}$$

$$m_k^{(s,n)} = \frac{C_{k-1}^n}{(k-1)!} \sum_{l=s}^{k-1} (-1)^{l-s} C_k^{l-s} (k-1-l)^{k-1-n} \qquad a, s, n \in \{0, \dots, k-1\}, \tag{2.35}$$

where the scalar $C_j^i = \frac{j!}{i!(j-i)!}$. Substituting $\boldsymbol{\lambda}(u) = \tilde{\mathbf{M}}_k \mathbf{u}$ into Equation (2.33) results in

$$\mathbf{p}(u) = \mathbf{p}_i + \sum_{j=1}^{k-1} \boldsymbol{\lambda}_j(u) \mathbf{d}_j^i. \tag{2.36}$$

Equation (2.36) describes the smooth translation of a rigid body in continuous time (see Figure 2.2 for an example). While the development above focuses on vector space splines, B-splines can also be defined on Lie groups, including the group $\mathrm{SO}(3)$ of rotations,

$$\mathbf{R}(u) = \mathbf{R}_i \prod_{j=1}^{k-1} \exp\left(\boldsymbol{\lambda}_j(u) \boldsymbol{\phi}_j^{i\,\wedge}\right), \tag{2.37}$$

where $\mathbf{R}_i$ is a control point of the rotation spline and $\boldsymbol{\phi}_j^i = \log(\mathbf{R}_{i+j-1}^{\mathsf{T}} \mathbf{R}_{i+j})$. Using this formulation of the rotation B-spline, the recursive definition of the rotational velocity is

$$\boldsymbol{\omega}_{(j)}(u) = \exp\left(-\boldsymbol{\lambda}_{j-1}(u) \boldsymbol{\phi}_{j-1}^{i\,\wedge}\right) \boldsymbol{\omega}_{(j-1)} + \dot{\boldsymbol{\lambda}}_{j-1}(u) \boldsymbol{\phi}_{j-1}^i, \tag{2.38}$$

$$\boldsymbol{\omega}_{(1)}(u) = \mathbf{0}, \tag{2.39}$$

where $\boldsymbol{\omega}_{(k)}$ is the rotational velocity defined by the $\mathrm{SO}(3)$ B-spline of order k. Further, the time derivative $\dot{\mathbf{R}}$ is given by $\dot{\mathbf{R}}(u) = \mathbf{R}(u) \boldsymbol{\omega}_{(k)}(u)^\wedge$. Finally, the recursive definition of the rotational acceleration is

$$\dot{\boldsymbol{\omega}}_{(j)}(u) = \dot{\boldsymbol{\lambda}}_{j-1}(u) \boldsymbol{\omega}_{(j)}{}^\wedge \boldsymbol{\phi}_{j-1}^i + \exp\left(-\boldsymbol{\lambda}_{j-1}(u) \boldsymbol{\phi}_{j-1}^{i\,\wedge}\right) \dot{\boldsymbol{\omega}}_{(j-1)} + \ddot{\boldsymbol{\lambda}}_{j-1}(u) \boldsymbol{\phi}_{j-1}^i, \tag{2.40}$$

$$\dot{\boldsymbol{\omega}}_{(1)}(u) = \mathbf{0}, \tag{2.41}$$

where $\dot{\boldsymbol{\omega}}_{(k)}$ is the rotational acceleration of the rotation matrix defined by a $\mathrm{SO}(3)$ B-spline of order k.

Since the control points of the $\mathrm{SO}(3)$ B-spline are elements of $\mathrm{SO}(3)$, we must use the perturbation approach from Section 2.1.3 to compute Jacobians with respect to the control points. Following our left-perturbation framework, the perturbed control point differences $\boldsymbol{\phi}_j^i$ and $\boldsymbol{\phi}_{j+1}^i$ are

$$\boldsymbol{\phi}_j^i = \log\left(\exp\left(\boldsymbol{\phi}_j^{i\,\wedge}\right) \exp\left(\left(\mathbf{R}_{i+j}{}^{\mathsf{T}} \boldsymbol{\psi}_{i+j}\right)^\wedge\right)\right)^\vee, \tag{2.42}$$

$$\boldsymbol{\phi}_{j+1}^i = \log\left(\exp\left(\boldsymbol{\phi}_{j+1}^{i\,\wedge}\right) \exp\left(-\left(\mathbf{R}_{i+j+1}{}^{\mathsf{T}} \boldsymbol{\psi}_{i+j}\right)^\wedge\right)\right)^\vee, \tag{2.43}$$

respectively. Since the magnitude of $\boldsymbol{\psi}_{i+j}$ is small, we can use an approximation of the Baker-Campbell-

Hausdorff formula to simplify Equations (2.42) and (2.43). More specifically, we can use

$$\log\left(\exp\left(\boldsymbol{\phi}^{\wedge}\right)\exp\left(\boldsymbol{\eta}\left(\boldsymbol{\psi}\right)^{\wedge}\right)\right)^{\vee} \approx \boldsymbol{\phi} + \mathbf{J}_r^{-1}(\boldsymbol{\phi})\boldsymbol{\eta}(\boldsymbol{\psi}) \tag{2.44}$$

where

$$\mathbf{J}_r^{-1}(\boldsymbol{\phi}) = \frac{\phi}{2}\cot\frac{\phi}{2}\mathbf{I} + \left(1 - \frac{\phi}{2}\cot\frac{\phi}{2}\right)\mathbf{a}\mathbf{a}^{\mathsf{T}} + \frac{\phi}{2}\mathbf{a}^{\wedge}, \tag{2.45}$$

and $\boldsymbol{\eta} : \mathbb{R}^3 \to \mathbb{R}^3$, and $\boldsymbol{\eta}(\boldsymbol{\psi})$ is small. The Jacobians of the control point differences $\boldsymbol{\phi}_j^i$ and $\boldsymbol{\phi}_{j+1}^i$ with respect to the control point perturbation, $\boldsymbol{\psi}_{i+j}$, are

$$\frac{\partial \boldsymbol{\phi}_j^i}{\partial \boldsymbol{\psi}_{i+j}} = \mathbf{J}_r^{-1}(\boldsymbol{\phi}_j^i)\mathbf{R}_{i+j}{}^{\mathsf{T}}, \tag{2.46}$$

$$\frac{\partial \boldsymbol{\phi}_{j+1}^i}{\partial \boldsymbol{\psi}_{i+j}} = -\frac{\partial \boldsymbol{\phi}_{j+1}^i}{\partial \boldsymbol{\psi}_{i+j+1}}. \tag{2.47}$$

Given $t_i \leq t < t_{i+1}$ and the normalized time u, the Jacobians of the function $\mathbf{f}(\mathbf{R}_i, \ldots, \mathbf{R}_{i+k-1}, t)$ with respect to the perturbations of the control points $\boldsymbol{\psi}_i$ and $\boldsymbol{\psi}_{i+j}$ are

$$\frac{\partial \mathbf{f}(\mathbf{R}_i, \ldots, \mathbf{R}_{i+k-1}, t)}{\partial \boldsymbol{\psi}_{i+j}} = \frac{\partial \mathbf{f}}{\partial \boldsymbol{\phi}_j^i}\frac{\partial \boldsymbol{\phi}_j^i}{\partial \boldsymbol{\psi}_{i+j}} + \frac{\partial \mathbf{f}}{\partial \boldsymbol{\phi}_{j+1}^i}\frac{\partial \boldsymbol{\phi}_{j+1}^i}{\partial \boldsymbol{\psi}_{i+j}} \quad \forall\, j > 0 \tag{2.48}$$

$$\frac{\partial \mathbf{f}(\mathbf{R}_i, \ldots, \mathbf{R}_{i+k-1}, t)}{\partial \boldsymbol{\psi}_i} = \frac{\partial \mathbf{f}}{\partial \boldsymbol{\psi}_i} + \frac{\partial \mathbf{f}}{\partial \boldsymbol{\phi}_1^i}\frac{\partial \boldsymbol{\phi}_1^i}{\partial \boldsymbol{\psi}_i}. \tag{2.49}$$

Consequently, to determine the Jacobian of a function with respect to the perturbations of the control points, we only need to compute it with respect to the control point $\mathbf{R}_i$ and each control point difference $\boldsymbol{\phi}_j^i$. Finally, we note that Equations (2.38) and (2.40) only depend on $\mathbf{R}_i$ through $\boldsymbol{\phi}_1^i$, so the Jacobians of these quantities with respect to $\boldsymbol{\psi}_i$ do not require $\frac{\partial \mathbf{f}}{\partial \boldsymbol{\psi}_i}$.

2.2 Probability Theory

All real-world sensor measurements are noisy. We model noise as a random variable that follows a known probability distribution. A random vector $\mathbf{n}$, drawn from a multivariate Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$, is denoted by

$$\mathbf{n} \sim \mathcal{N}\left(\boldsymbol{\mu}, \boldsymbol{\Sigma}\right). \tag{2.50}$$

While our sensor measurement models often include an additive, zero-mean Gaussian noise term, $\mathbf{n} \sim \mathcal{N}\left(\mathbf{0}, \boldsymbol{\Sigma}\right)$, which is useful for vector space measurements, rotation matrices are elements of $\mathrm{SO}(3)$ and therefore we cannot model $\mathrm{SO}(3)$ measurement noise as an additive noise term. One probabilistic measurement model for $\mathrm{SO}(3)$ is

$$\mathbf{R} \sim \exp\left(\mathbf{n}^{\wedge}\right)\bar{\mathbf{R}}, \quad \mathbf{n} \sim \mathcal{N}\left(\mathbf{0}, \boldsymbol{\Sigma}\right), \tag{2.51}$$

where $\bar{\mathbf{R}}$ is a noise-free, mean rotation and $\mathbf{n}$ is a noisy perturbation. In this case, we use the *left* rotation perturbation. Significantly, both the middle and right perturbations are also valid.

Estimation problems that model measurement noise as Equation (2.51) cannot be relaxed into convex optimization problems. To enable convex relaxations, we represent uncertainty over $\mathrm{SO}(d)$ using the Langevin

distribution

$$\mathbf{R} \sim \text{Lang}(\bar{\mathbf{R}}, \kappa), \tag{2.52}$$

where $\text{Lang}(\mathbf{A}, \kappa)$ denotes the Langevin distribution with mode $\mathbf{A}$ and concentration parameter κ. The density function for this distribution is

$$p\left(\mathbf{R}; \bar{\mathbf{R}}, \kappa\right) = \frac{1}{c\left(\kappa\right)} \exp\left(\kappa \operatorname{tr}\left(\bar{\mathbf{R}}^{\mathsf{T}}\mathbf{R}\right)\right), \tag{2.53}$$

where $c\left(\kappa\right)$ is a normalizing factor. To sample $\mathbf{R}$, we perturb the mode $\bar{\mathbf{R}}$ with a rotation about a uniformly-drawn axis $\mathbf{a} \sim \mathcal{U}\left(\mathrm{S}^2\right)$ with a magnitude drawn from a von Mises distribution $\theta \sim \text{vM}\left(0, 2\kappa\right)$. In this case, the mode of the von Mises distribution is 0. When the concentration parameter $\kappa = 0$, the von Mises distribution becomes a uniform distribution on $(-\pi, \pi]$. As the concentration parameter κ approaches ∞, the von Mises distribution converges to a zero-mean Gaussian distribution with covariance κ^{-1}. We refer readers to Rosen et al. [2019] for additional information on the Langevin distribution and some related identities.

2.3 Intrinsic, Extrinsic, and Temporal Calibration

Calibration parameters can be separated (roughly) into three classes: intrinsic, extrinsic, and temporal. We use the set of estimated parameters to name or designate various calibration algorithms. However, an algorithm may estimate parameters from multiple classes simultaneously. In these cases, the name of the algorithm is a compound of the estimated parameter classes.

For a given sensor measurement model, the *intrinsic* parameters define internal characteristics that vary from sensor to sensor. Inaccurate intrinsic parameters cause discrepancies between the model and raw data, degrading the performance of downstream algorithms. The parameters may vary for many reasons, such as manufacturing variance and operating temperature, so we require a process known as intrinsic calibration to determine these parameters. For example, Szeliski [2022] summarizes various intrinsic camera calibration methods to determine camera-intrinsic and lens-distortion parameters. Further, Petrov et al. [2021] presents an intrinsic calibration scheme to estimate the complex coefficients of each antenna within a radar unit. In Chapters 5 and 8, we calibrate the camera-intrinsic parameters using the algorithm from Oth et al. [2013] and assume that the factory-estimated radar-intrinsic parameters are sufficiently accurate.

When fusing data from multiple sensors, we require the rigid-body transformation (i.e., spatial parameters) between each sensor coordinate frame in order to express the data in one (common) coordinate frame. If we express the data from multiple sensors in a single coordinate frame and the spatial parameters are inaccurate, then mutually observed features may appear spatially shifted, which biases downstream algorithms. User modifications and wear and tear may also change the spatial parameters. Consequently, we apply *extrinsic* calibration algorithms (i.e., relative to the sensors) to estimate the spatial parameters. Depending on the sensor measurements, simultaneously estimating rotation and translation can be challenging, so some procedures solve for the rotation and translation parameters separately (or only for the rotation parameters). Chapters 5 and 7 present extrinsic calibration methods for radar-camera, radar-radar, and camera-pose sensor pairs, while Chapter 8 presents an extrinsic calibration method for systems with arbitrarily many cameras.

Most modern electronic sensors contain internal circuitry to preprocess raw sensor measurements into a form that can be transmitted to a host (e.g., CPU). Preprocessing introduces some latency; in turn, if a sensor includes an internal clock to timestamp each measurement, the assigned timestamp may not accurately reflect the true time at which the measured event occurred. Further, if one sensor clock is not synchronized to other

clocks (via, e.g., hardware triggering) then arbitrary transmission delays may also lead to inaccurate measurement timestamps upon arrival at the host. In this case, we say that measurement timestamps are, relatively, temporally offset. Since power cycling or reconfiguring a system may change the temporal offsets, we require *temporal* calibration to estimate these values. Chapter 5 presents a method to simultaneously estimate the spatial and temporal calibration parameters of a radar-camera pair, in a process known as *spatiotemporal calibration*.

2.4 Homography, Reprojection, and Motion Models for Calibration

Calibration algorithms can also be classified (again, roughly) based on homography, reprojection, or motion, depending on the model used to relate the sensor measurements. Homography-based methods estimate a linear map between two 2D planes. Reprojection-based methods determine the calibration parameters by aligning mutually observed 'features,' where a feature is an identifiable point in the environment (e.g., a checkerboard corner). Motion-based methods estimate calibration parameters by relating the estimated motion of each sensor using rigid-body kinematic constraints.

In homography-based algorithms, each sensor detects identifiable features that lie on a plane. For example, Sugimoto et al. [2004] use this approach to calibrate the 2D radar-camera system (see Figure 2.3). Assuming that each sensor produces a 2D measurement, $\mathbf{q}_i$ and $\mathbf{p}_i$, of landmark l_i, two possible models are

$$\underbrace{\begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix}}_{\mathbf{q}_i} = \underbrace{\begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ 0 & 0 & 1 \end{bmatrix}}_{\mathbf{H}_a} \underbrace{\begin{bmatrix} r_i^x \\ r_i^z \\ 1 \end{bmatrix}}_{\mathbf{p}_i}, \tag{2.54}$$

$$\underbrace{\begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix}}_{\mathbf{q}_i} = \underbrace{\begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}}_{\mathbf{H}_p} \underbrace{\begin{bmatrix} r_i^x \\ r_i^z \\ 1 \end{bmatrix}}_{\mathbf{p}_i}, \tag{2.55}$$

where $\mathbf{H}_a$ and $\mathbf{H}_p$ are affine and projective (homography) matrices, respectively.

Critically, requiring the sensors to detect features on a plane requires the sensors to have overlapping fields of view, which can restrict potential system configurations and calibration venues. Further details on the applications and limitations of homography-based calibration for radar-camera systems are given in Section 5.1.1.

Reprojection-based methods estimate calibration parameters by aligning mutually observed features that

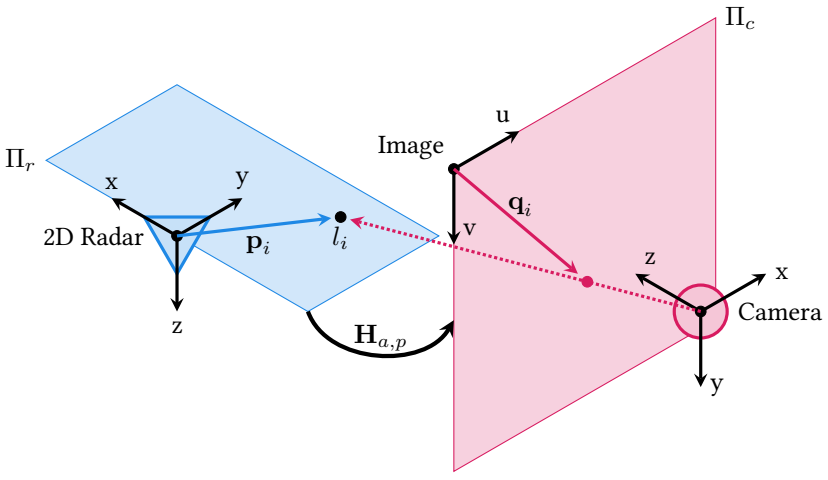


Figure 2.3: Illustration of the 2D radar-camera homography-based calibration. Given a landmark l_i, the matrices $\mathbf{H}_a$ and $\mathbf{H}_p$ map the (homogeneous) coordinates of the point $\mathbf{p}_i$ of landmark on the radar sensing plane, Π_r, to the homogeneous pixel coordinate, $\mathbf{q}_i$, of the landmark on the image plane, Π_c.

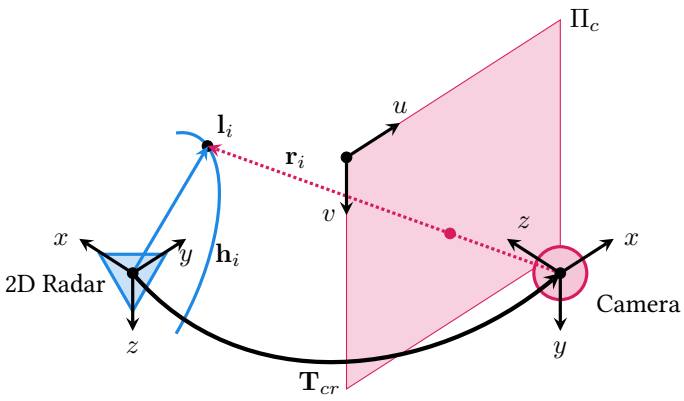


Figure 2.4: Illustration of a 2D radar-camera reprojection model. Given a landmark l_i, the matrix $\mathbf{T}_{cr}$ aligns the radar arc, $\mathbf{h}_i$, and the camera ray, $\mathbf{r}_i$.

are detected and matched. As shown in Figure 2.4, reprojection-based methods determine the parameters that minimize the distance between pairs of features, as measured in each sensor's reference frame. Often, these methods rely on specialized targets to enable easy feature detection and correspondence identification. For calibration to be successful, the targets must be viewed from a number of locations relative to the sensors; this can be achieved by moving a single target or by placing multiple targets at different locations. Further, sensor platform motion may provide different views of a target, but achieving a sufficient diversity of views can be challenging with underactuated systems (e.g., AVs). The details of reprojection-based models depend on the specialized targets, sensing modalities, and system configurations involved. Overlapping fields of view are still required. In Section 5.1.2, we provide more details on reprojection-based calibration for systems where one sensor is a radar unit.

Unlike methods based on homography and reprojection, motion-based calibration leverages knowledge of the pose or velocity of each sensor to determine the calibration parameters. In particular, measurements from some sensors can be used to determine the pose or ego-velocity of the sensor itself relative to an inertial coordinate frame. As shown in Figure 2.5, motion-based algorithms use rigid-body kinematic constraints to estimate, for example, the relative pose between sensors. Importantly, these algorithms do not typically require overlapping fields of view or specialized targets, enabling calibration in a wider range of environments. However, the system must undergo sufficient excitation, which may be challenging for underactuated systems.

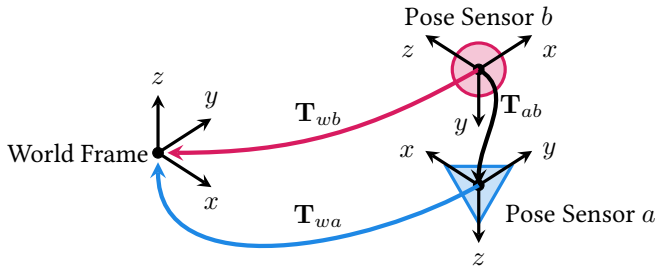


Figure 2.5: Illustration of motion-based calibration. Each sensor enables estimation the sensor pose relative to the world frame $\underrightarrow{\mathcal{F}}_w$. Given the poses from $\underrightarrow{\mathcal{F}}_w$ to the references frames of sensors a and b, $\underrightarrow{\mathcal{F}}_a$ and $\underrightarrow{\mathcal{F}}_b$, respectively, the matrix $\mathbf{T}_{ab}$ is the transform that best describes the rigid-body connection between the self-estimated poses of the sensors.

2.5 Sensor Abstractions

Often, raw measurements from an intrinsically-calibrated sensor can be used to estimate the pose or velocity of the sensor itself relative to an inertial frame. For example, the pose of stereo cameras can be found from raw measurements of static visual features in the environment. Notably, although sensing modalities may differ, the 'derived' measurements of pose or motion are the same. Calibration algorithms that leverage these derived measurement models are applicable to a wider range of systems, because the algorithms do not rely on the underlying sensing modalities specifically. In this thesis, we will call these types of derived measurement models *sensor abstractions*. Below, we review three such abstractions: pose, scaled pose, and ego-velocity. In each case, we label the sensor reference frame as $\underrightarrow{\mathcal{F}}_s$ and the stationary world frame as $\underrightarrow{\mathcal{F}}_w$.

2.5.1 Pose Sensors

When raw sensor measurements enable estimation of the sensor pose relative to a stationary (inertial) reference frame, we can define a *pose sensor abstraction*. Importantly, stereo cameras, lidars, and depth cameras all fall into the category of pose sensors. Two possible pose sensor measurement models are

$$\mathbf{T}_{ws} = \begin{bmatrix} \mathbf{R}_{ws}\exp\left(\mathbf{n}_r{}^{\wedge}\right) & \mathbf{t}_w^{sw} + \mathbf{n}_t \\ \mathbf{0}^{\intercal} & 1 \end{bmatrix}, \tag{2.56}$$

$$\mathbf{T}_{sw} = \begin{bmatrix} \exp\left(\mathbf{n}_r{}^{\wedge}\right)\mathbf{R}_{sw} & \mathbf{t}_s^{ws} + \mathbf{n}_t \\ \mathbf{0}^{\intercal} & 1 \end{bmatrix}, \tag{2.57}$$

where $\mathbf{n}_r \sim \mathcal{N}\left(\mathbf{0}, \mathbf{\Sigma}_r\right)$ and $\mathbf{n}_t \sim \mathcal{N}\left(\mathbf{0}, \mathbf{\Sigma}_t\right)$.

Importantly, the assumption that the noise terms $\mathbf{n}_r$ and $\mathbf{n}_t$ are Gaussian-distributed may not hold in practice as they are approximations. The abstracted pose measurement is derived from raw sensor measurements that are usually Gaussian-distributed, but the mapping may be nonlinear and hence the resulting noise terms may be non-Gaussian. However, in these cases, we approximate the non-Gaussian noise by a Gaussian distribution.

2.5.2 Scaled Pose Sensors

When raw sensor measurements enable estimation of the sensor pose up to an unknown translational scale, we can define a *scaled pose sensor abstraction*. As shown in Chiuso et al. [2002], monocular cameras are scaled pose sensors, if (and only if) the camera observes the same features across time to ensure that the scale factor remains consistent. Two possible scaled pose measurement models are

$$\mathbf{T}_{ws} = \begin{bmatrix} \mathbf{R}_{ws}\exp(\mathbf{n}_r) & \gamma\mathbf{t}_w^{sw} + \mathbf{n}_t \\ \mathbf{0}^{\intercal} & 1 \end{bmatrix}, \tag{2.58}$$

$$\mathbf{T}_{sw} = \begin{bmatrix} \exp(\mathbf{n}_r)\mathbf{R}_{sw} & \gamma\mathbf{t}_s^{ws} + \mathbf{n}_t \\ \mathbf{0}^{\intercal} & 1 \end{bmatrix}, \tag{2.59}$$

where γ is the unknown scale factor, $\mathbf{n}_r \sim \mathcal{N}\left(\mathbf{0}, \mathbf{\Sigma}_r\right)$ and $\mathbf{n}_t \sim \mathcal{N}\left(\mathbf{0}, \mathbf{\Sigma}_t\right)$. As with pose sensors, the noise distribution is an approximation of a non-Gaussian distribution.

2.5.3 Ego-Velocity Sensors

If the raw measurements from a sensor enable estimation of its instantaneous velocity relative to an inertial frame, we can define an *ego-velocity* sensor abstraction. Radars units are currently the most popular such sensors, but some lidars can also be considered as ego-velocity sensors.

An ego-velocity measurement model may generally have one of three forms. Typically, its sensor model is

$$\begin{aligned} \mathbf{v}_s^{sw}(t) &= \mathbf{R}_{ws}(t)^T \frac{\partial \mathbf{t}_w^{sw}(t)}{\partial t} + \mathbf{n}_v, \\ \mathbf{n}_v &\sim \mathcal{N}\left(\mathbf{0}_{3\times 1}, \boldsymbol{\Sigma}_v\right). \end{aligned} \tag{2.60}$$

We may also model the ego-velocity measurement relative to the moving reference frame of the sensor instead of the stationary world frame. This *robocentric* formulation is

$$\mathbf{v}_s^{sw}(t) = -\dot{\mathbf{t}}_s^{ws}(t) - \boldsymbol{\omega}_s^{sw}(t)^{\wedge}\mathbf{t}_s^{ws}(t) + \mathbf{n}_v. \tag{2.61}$$

Lastly, when fusing ego-velocity measurements with other data, we may wish to describe them relative to another rigidly-connected sensor reference frame. Given a rigidly connected frame $\underrightarrow{\mathcal{F}}_a$, the ego-velocity measurement model for the sensor frame s is

$$\mathbf{v}_s^{sw}(t) = \mathbf{R}_{as}{}^{\mathsf{T}} \begin{bmatrix} \mathbf{I}_{3\times 3} & -\mathbf{t}_a^{sa\wedge} \end{bmatrix} \begin{bmatrix} \mathbf{v}_a^{aw}(t) \\ \boldsymbol{\omega}_a^{aw}(t) \end{bmatrix} + \mathbf{n}_v, \tag{2.62}$$

where $\mathbf{R}_{as}$ and $\mathbf{t}_a^{sa}$ are the rotation and translation (moment arm) between the two sensor coordinate frames. In this case, it is important to consider the moment arm so as to properly account for the linear component of the velocity that is induced by the rotational velocity of the system.

2.6 Summary

In this chapter, we reviewed the mathematical definitions and concepts important to formulating the calibration problems addressed in this thesis. We discussed mathematical representations of translations, rotations, and rigid-body transformations, and provided some of the associated Jacobians. Having reviewed mathematical preliminaries, we then defined the probability distributions that are involved in the calibration problems in Chapters 5 to 8. To position our work, we provided a brief taxonomy of calibration problems (intrinsic, extrinsic, and temporal). We introduced motion-based calibration, which is the focus of this thesis, and compared this method to homography- and reprojection-based calibration. Finally, we presented compact descriptions of three sensor abstractions that enable our calibration algorithms to operate with a wide variety of systems.

Chapter 3

Optimization

Typically, calibration algorithms leverage batch optimization to determine the calibration values that 'best' explain the measurements. In the context of estimation, batch optimization involves the use of all available data to estimate of the system parameters of interest. This approach is in contrast to (sequential) filtering, where new data are processed as they arrive. The batch approach has some advantages over filtering, including, for example, an ability to capture dependencies between measurements and parameters across time, but these advantages come with a greater computational cost.

This chapter reviews batch nonlinear and convex optimization. Section 3.1 discusses the standard probabilistic formulation of nonlinear least squares optimization. Section 3.2 details a method to solve certain batch calibration problems through convex optimization.

3.1 Nonlinear Batch Estimation

Generally, we formulate calibration processes for autonomous systems as optimization problems. While the exact form of an optimization problem is application-specific, this section presents, at a high level, a general framework for structuring and solving batch optimization problems in robotics. In particular, we review a general formulation of discrete, nonlinear batch optimization problems and the Gauss-Newton optimization algorithm.

3.1.1 System Models and Error Distributions

In general, for the problems we consider, the motion and measurement models will be

$$\mathbf{x}(t_i) = \mathbf{f}(\mathbf{x}(t_{i-1}), \mathbf{u}(t_i), \mathbf{w}(t_i)), \tag{3.1}$$

$$\mathbf{y}(t_i) = \mathbf{h}(\mathbf{x}(t_i), \mathbf{n}(t_i)), \tag{3.2}$$

respectively, at time t_i, where $\mathbf{x}(t) \in \mathbb{R}^n$ is the system state vector, $\mathbf{y}(t) \in \mathbb{R}^m$ is the measurement vector, and $\mathbf{u}(t) \in \mathbb{R}^p$ is the input or control vector. The functions $\mathbf{f}(\cdot)$ and $\mathbf{h}(\cdot)$ are usually nonlinear. The vectors $\mathbf{w}(t_k)$ and $\mathbf{n}(t_k)$ are the process and measurement noise terms, respectively. While the noise vectors are not restricted to any particular form, for simplicity, we model random noise as zero-mean and Gaussian-distributed, that is, $\mathbf{w}(t_i) \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_f(t_i))$ and $\mathbf{n}(t_i) \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_y(t_i))$. Further, we may have a prior belief about the state, which

we model as

$$\mathbf{x}(t_0) = \check{\mathbf{x}}_0 + \mathbf{w}_0, \tag{3.3}$$

where $\check{\mathbf{x}}_0$ is our state prior and $\mathbf{w}_0 \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_0)$. As we show below, the cost function of our optimization problem is formed by the measurement, state prior, and motion error distributions. The error distributions at time t_i are

$$\mathbf{e}_{u,i}(\mathbf{x}(t_{i-1}), \mathbf{x}(t_i)) = \mathbf{f}(\mathbf{x}(t_{i-1}), \mathbf{u}(t_i), \mathbf{w}(t_i)) - \mathbf{x}(t_i), \tag{3.4}$$

$$\mathbf{e}_{y,i}(\mathbf{x}(t_i)) = \mathbf{y}(t_i) - \mathbf{h}(\mathbf{x}(t_i), \mathbf{n}(t_i)), \tag{3.5}$$

$$\mathbf{e}_p(\mathbf{x}(t_0)) = \check{\mathbf{x}}_0 - \mathbf{x}(t_0) + \mathbf{w}_0. \tag{3.6}$$

Given our Gaussian noise assumption, the error distributions become

$$\mathbf{e}_{u,i}(\mathbf{x}(t_{i-1}), \mathbf{x}(t_i)) = \mathbf{f}(\mathbf{x}(t_{i-1}), \mathbf{u}(t_i), t_i - t_{i-1}) - \mathbf{x}(t_i) \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_f(t_i)), \tag{3.7}$$

$$\mathbf{e}_{y,i}(\mathbf{x}(t_i)) = \mathbf{y}(t_i) - \mathbf{h}(\mathbf{x}(t_i)) \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_y(t_i)), \tag{3.8}$$

$$\mathbf{e}_p(\mathbf{x}(t_0)) = \check{\mathbf{x}}_0 - \mathbf{x}(t_0) \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_0). \tag{3.9}$$

3.1.2 Pose and Scaled Pose Sensor Models and Error Distributions

For both the pose- and scaled-pose-sensor abstractions, the derived measurement is an element of $\mathrm{SE}(3)$. We split the derived measurement model into translation and rotation components. In general, derived translation error distributions take the form of Equation (3.8), but modelling the derived rotation measurement is more complicated. The rotation measurement model and error distribution take the form

$$\mathbf{R}_y(t_i) = \exp(\mathbf{n}^\wedge)\, \mathbf{H}(\mathbf{x}(t_i)), \tag{3.10}$$

$$\mathbf{e}_{y,i}(\mathbf{x}(t_i)) = \log(\mathbf{R}_y(t_i)\mathbf{H}^\mathsf{T}(\mathbf{x}(t_i)))^\vee \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_y(t_i)), \tag{3.11}$$

where $\mathbf{R}_y(t_i) \in \mathrm{SO}(3)$ is the measurement and $\mathbf{n} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_y(t_i))$. The function $\mathbf{H}(\cdot)$ is a nonlinear function that maps to $\mathrm{SO}(3)$. We model measurement noise using the left Lie algebraic perturbation scheme discussed in Section 2.2. In optimization problems that leverage these rotation models and error distributions, $\mathbf{x}(t_i)$ contains elements of $\mathrm{SO}(3)$. Since $\mathrm{SO}(3)$ is not a vector space, we refer to $\mathbf{x}(t_i)$ as the state set instead of the state vector.

3.1.3 Probabilistic Optimization Problems

Currently, our models only describe the state at a single instance of time, but batch optimization leverages a sequence of measurements across time. To model a sequence of measurements, we formulate the optimization problem using a discrete representation of the time-varying state vector or set. If we wish to estimate the state from time t_0 to t_{M-1}, then we only require the state at the M measurement times. As a result, we can use a discrete representation of the state vector, that is,

$$\mathbf{x}^\mathsf{T} = \begin{bmatrix} \mathbf{x}(t_0)^\mathsf{T} & \cdots & \mathbf{x}(t_{M-1})^\mathsf{T} \end{bmatrix}, \tag{3.12}$$

or of the state set, that is

$$\mathbf{x} = \{\mathbf{x}(t_0), \mathbf{x}(t_1), \ldots, \mathbf{x}(t_{M-1})\}. \tag{3.13}$$

Here, the time $t_{i+1} = t_i + \Delta t$ and Δt is the time between measurements.

Using the measurement, state prior, and motion error distributions, we can formulate a maximum a posteriori (MAP) optimization problem:

$$\mathbf{x}^* = \underset{\mathbf{x}}{\operatorname{argmax}}\, p(\mathbf{x} \mid \mathbf{u}, \mathbf{y}), \tag{3.14}$$

where $\mathbf{x}$ is either Equation (3.12) or (3.13). Further, we have

$$\mathbf{y}^\intercal = \begin{bmatrix} \mathbf{y}(t_0)^\intercal & \cdots & \mathbf{y}(t_{M-1})^\intercal \end{bmatrix}, \tag{3.15}$$

$$\mathbf{u}^\intercal = \begin{bmatrix} \mathbf{u}(t_0)^\intercal & \cdots & \mathbf{u}(t_{M-1})^\intercal \end{bmatrix}. \tag{3.16}$$

Applying Bayes' rule, the problem becomes finding $\mathbf{x}^*$ such that

$$\mathbf{x}^* = \underset{\mathbf{x}}{\operatorname{argmax}}\, p(\mathbf{y} \mid \mathbf{x}) p(\mathbf{x} \mid \mathbf{u}). \tag{3.17}$$

Assuming all error are statistically independent, we can formulate the optimization problem as

$$\mathbf{x}^* = \underset{\mathbf{x}}{\operatorname{argmax}}\, p(\mathbf{x}(t_0) \mid \check{\mathbf{x}}_0) \prod_{i=1}^{M-1} p(\mathbf{x}(t_i) \mid \mathbf{x}(t_{i-1}), \mathbf{u}(t_i)) \prod_{i=0}^{M-1} p(\mathbf{y}(t_i) \mid \mathbf{x}(t_i)). \tag{3.18}$$

After taking the negative log of the cost function, the problem becomes

$$\mathbf{x}^* = \underset{\mathbf{x}}{\operatorname{argmax}} \left(-\ln p(\mathbf{x}(t_0) \mid \check{\mathbf{x}}_0) - \sum_{i=1}^{N} \ln p(\mathbf{x}(t_i) \mid \mathbf{x}(t_{i-1}), \mathbf{u}(t_i)) - \sum_{i=0}^{N} \ln p(\mathbf{y}(t_i) \mid \mathbf{x}(t_i)) \right), \tag{3.19}$$

Since the errors are zero-mean and Gaussian distributed, our MAP problem takes the following form.

Problem 1. *Nonlinear MAP Optimization Problem*

$$\min_{\mathbf{x}} \sum_{i=0}^{N} (J_{u,i}(\mathbf{x}) + J_{y,i}(\mathbf{x})), \tag{3.20}$$

where

$$J_{u,i}(\mathbf{x}) = \begin{cases} \frac{1}{2} \mathbf{e}_p(\mathbf{x})^\intercal \boldsymbol{\Sigma}_0^{-1} \mathbf{e}_p(\mathbf{x}), & i = 0, \\ \frac{1}{2} \mathbf{e}_{u,i}(\mathbf{x})^\intercal \boldsymbol{\Sigma}_f(t_i)^{-1} \mathbf{e}_{u,i}(\mathbf{x}), & i > 0, \end{cases} \tag{3.21}$$

$$J_{y,i}(\mathbf{x}) = \frac{1}{2} \mathbf{e}_{y,i}(\mathbf{x})^\intercal \boldsymbol{\Sigma}_y(t_i)^{-1} \mathbf{e}_{y,i}(\mathbf{x}) \; \forall \; i = 0, \ldots N. \tag{3.22}$$

Notably, when we do not have a prior, the MAP problem reduces to a maximum likelihood estimation (MLE) problem.

Problem 2. *Nonlinear MLE Problem*

$$\min_{\mathbf{x}} \sum_{i=0}^{N} J_{y,i}(\mathbf{x}) + \sum_{i=1}^{N} J_{u,i}(\mathbf{x}), \tag{3.23}$$

where

$$J_{u,i}(\mathbf{x}) = \frac{1}{2}\mathbf{e}_{u,i}(\mathbf{x})^\intercal \boldsymbol{\Sigma}_f(t_i)^{-1}\mathbf{e}_{u,i}(\mathbf{x}), \tag{3.24}$$

$$J_{y,i}(\mathbf{x}) = \frac{1}{2}\mathbf{e}_{y,i}(\mathbf{x})^\intercal \boldsymbol{\Sigma}_y(t_i)^{-1}\mathbf{e}_{y,i}(\mathbf{x}). \tag{3.25}$$

3.1.4 Least-Squares Batch Optimization

There are many possible ways to solve Problem 2. The popular approach that used in this study is the Gauss-Newton algorithm. Before discussing the Gauss-Newton solver, we simplify Equation (3.23) by defining

$$\mathbf{e}(\mathbf{x}) = \begin{bmatrix}\mathbf{e}_y & \mathbf{e}_u\end{bmatrix} \tag{3.26}$$

$$\mathbf{e}_y{}^\intercal = \begin{bmatrix}\mathbf{e}_{y,0} & \cdots & \mathbf{e}_{y,N}\end{bmatrix}^\intercal, \tag{3.27}$$

$$\mathbf{e}_u{}^\intercal = \begin{bmatrix}\mathbf{e}_{u,1} & \cdots & \mathbf{e}_{u,N}\end{bmatrix}^\intercal, \tag{3.28}$$

$$\boldsymbol{\Sigma} = \text{blkdiag}(\boldsymbol{\Sigma}_y, \boldsymbol{\Sigma}_f), \tag{3.29}$$

$$\boldsymbol{\Sigma}_y = \text{blkdiag}(\boldsymbol{\Sigma}_y(t_0), \ldots, \boldsymbol{\Sigma}_y(t_N)), \tag{3.30}$$

$$\boldsymbol{\Sigma}_f = \text{blkdiag}(\boldsymbol{\Sigma}_f(t_1), \ldots, \boldsymbol{\Sigma}_f(t_N)). \tag{3.31}$$

The cost function then becomes

$$J(\mathbf{x}) = \mathbf{e}(\mathbf{x})^\intercal \boldsymbol{\Sigma}^{-1}\mathbf{e}(\mathbf{x}). \tag{3.32}$$

Since $\boldsymbol{\Sigma}$ is a symmetric positive definite matrix, we can decompose $\boldsymbol{\Sigma}^{-1} = \mathbf{L}^\intercal\mathbf{L}$ and define a modified error distribution

$$\mathbf{e}_{mod}(\mathbf{x}) = \mathbf{L}\mathbf{e}(\mathbf{x}) \tag{3.33}$$

Additionally, Equation (3.32) becomes

$$J(\mathbf{x}) = \mathbf{e}_{mod}(\mathbf{x})^\intercal \mathbf{e}_{mod}(\mathbf{x}). \tag{3.34}$$

In Gauss-Newton, we iteratively approximate the cost function $J(\mathbf{x})$ with a quadratic function and minimize the quadratic cost. With an initial guess, $\mathbf{x}_{op}$, for the state, we can linearize the modified error distribution with respect to the initial guess, that is

$$\mathbf{e}_{mod}(\mathbf{x}_{op} + \delta\mathbf{x}) \approx \mathbf{e}_{mod}(\mathbf{x}_{op}) + \left(\frac{\partial \mathbf{e}_{mod}(\mathbf{x})}{\partial \delta\mathbf{x}}\bigg|_{\mathbf{x}_{op}}\right)\delta\mathbf{x}. \tag{3.35}$$

The cost function becomes

$$J(\mathbf{x}_{op} + \delta\mathbf{x}) \approx \left(\mathbf{e}_{mod}(\mathbf{x}_{op}) + \left(\frac{\partial \mathbf{e}_{mod}(\mathbf{x})}{\partial \delta\mathbf{x}}\bigg|_{\mathbf{x}_{op}}\right)\delta\mathbf{x}\right)^\intercal \left(\mathbf{e}_{mod}(\mathbf{x}_{op}) + \left(\frac{\partial \mathbf{e}_{mod}(\mathbf{x})}{\partial \delta\mathbf{x}}\bigg|_{\mathbf{x}_{op}}\right)\delta\mathbf{x}\right), \tag{3.36}$$

which is our quadratic approximation of Equation (3.34). The minimum of $J(\mathbf{x}_{op} + \delta\mathbf{x})$ satisfies

$$\left(\frac{\partial \mathbf{e}_{mod}(\mathbf{x})}{\partial \delta\mathbf{x}}\bigg|_{\mathbf{x}_{op}}\right)^\intercal \left(\frac{\partial \mathbf{e}_{mod}(\mathbf{x})}{\partial \delta\mathbf{x}}\bigg|_{\mathbf{x}_{op}}\right)\delta\mathbf{x}^\star = -\left(\frac{\partial \mathbf{e}_{mod}(\mathbf{x})}{\partial \delta\mathbf{x}}\bigg|_{\mathbf{x}_{op}}\right)^\intercal \delta\mathbf{e}_{mod}(\mathbf{x}_{op}), \tag{3.37}$$

which we solve using an existing linear solver. We update our initial guess as

$$\mathbf{x}_{op} \leftarrow \mathbf{x}_{op} + \delta\mathbf{x}^{\star}. \tag{3.38}$$

We usually iterate this procedure until the magnitude of $\delta\mathbf{x}^{\star}$ is small. The solver converges to a nearby (local) minimum, so the initial guess for the state must be close to the global minimum of the problem. Otherwise, the solver may report a local minimum instead of the global minimum. Many modifications exist to improve the stability of this optimization approach and we refer interested readers to Barfoot [2024] and Agarwal et al. [2022].

In some cases, there may be states in $\mathbf{x}$ that are not elements of a vector space and that instead lie on curved manifolds. Up until now, our discussion of the Gauss-Newton algorithm has treated the state as belonging to a vector space. There are, however, many approaches to handling states that lie on curved manifolds. Similar to Equation (3.35), we choose to linearize the error equations about local left-perturbations of the state, hence the name on-manifold optimization. Specifically, we substitute $\mathbf{R}$ with $\exp\left(\boldsymbol{\psi}^{\wedge}\right)\mathbf{R}_{op}$ in the modified cost function. Since $\boldsymbol{\psi}$ is small, we use small-angle approximations, such as Equation (2.44), to achieve a modified cost function that is linear with respect to $\boldsymbol{\psi}$. Then, we can update any rotation matrices using $\mathbf{R}_{op} \leftarrow \exp\left(\boldsymbol{\psi}^{\star\wedge}\right)\mathbf{R}_{op}$ This approach allows us to solve what we refer to as a *generalized calibration problem.*

Problem 3. *Generalized Calibration Problem*

$$\min_{\mathbf{x}=\{\mathbf{R}_0,\ldots,\mathbf{R}_N,\mathbf{x}_v\}} J(\mathbf{x}), \tag{3.39}$$

$$\text{s.t. } \mathbf{R}_i \in \mathrm{SO}(d), \forall i = 1,\ldots,N, \tag{3.40}$$

where $\mathbf{x}_v \in \mathbb{R}^M$ and is the concatenation of all vector quantities in the state set.

3.2 Convex Batch Optimization

Previously, we noted that local solvers, such as Gauss-Newton, can find a local minimum instead of the global minimum. In this section, we present a subset of optimization problems where we can determine -or *certify*- whether the solution from our local solver is, in fact, the global minimum. Initially, we cast Problem 3 as a quadratically constrained quadratic program (QCQP). After reformulating the problem, we review convex sets, functions, and optimization problems (or convex *programs*). Unlike nonconvex optimization problems, a local minimum is the global minimum in a convex optimization problem. If we relax our nonconvex optimization problems into convex problems, then we can potentially certify the optimality of our solutions. Following our review, we discuss the Lagrangian dual of optimization problems, which enables convex relaxations of nonconvex problems. Finally, we derive two possible convex relaxations of a QCQP.

3.2.1 Quadratically Constrainted Quadratic Program

To demonstrate that Problem 3 can be a QCQP, we cast the cost function, $J(\mathbf{x})$, and rotation manifold constraints, $\mathbf{R}_i \in \mathrm{SO}(d), \forall i = 1,\ldots,N$, as homogeneous, quadratic functions (i.e., polynomials where each term with a nonzero coefficient has degree two). If the cost function is quadratic with respect to the state parameters, then

we can reformulate the cost function as

$$J(\mathbf{x}) = \mathbf{x}^\intercal \mathbf{A}\mathbf{x} + \mathbf{b}^\intercal \mathbf{x} + c, \tag{3.41}$$

where we modify $\mathbf{x}$ to be $\mathbf{x}^\intercal = \begin{bmatrix}\mathbf{x}_v^\intercal & \text{vec}\,(\mathbf{R}_0)^\intercal & \cdots & \text{vec}\,(\mathbf{R}_n)^\intercal\end{bmatrix}$. The matrix $\mathbf{A}$, vector $\mathbf{b}$, and scalar c are all fixed coefficients. Since Problem 3 is an optimization problem, we drop the fixed cost offset, c, and homogenize $J(\mathbf{x})$ by introducing the scalar variable $s \in \mathbb{R}$, where $s^2 = 1$. The cost function becomes

$$J(\mathbf{x}) = \mathbf{x}^\intercal \mathbf{Q}\mathbf{x}, \tag{3.42}$$

where we have appended s onto $\mathbf{x}$. Next, we reformulate the constraints in $\mathbf{R}_i \in \text{SO}(3)$ using the homogenization variable s, such that

$$\mathbf{R}_i \mathbf{R}_i^\intercal = s^2 \mathbf{I}, \tag{3.43}$$

$$\mathbf{R}_i^\intercal \mathbf{R}_i = s^2 \mathbf{I}, \tag{3.44}$$

$$\mathbf{R}_{i,(j,:)} \times \mathbf{R}_{i,(k,:)} = s\mathbf{R}_{i,(l,:)}, \;\; j,k,l \in \text{cyclic}(1,2,3). \tag{3.45}$$

For a given QCQP, each estimated rotation matrix introduces 21 constraints; six for row-orthogonality; six for column-orthogonality; and nine for cyclic cross-products. Additionally, we must introduce a homogenization variable and constraint if the cost function does not require homogenization. In general, the QCQP is as follows.

Problem 4. *Quadratically Constrained Quadratic Calibration Problem*

$$\min_{\mathbf{x}} \; \mathbf{x}^\intercal \mathbf{Q}\mathbf{x}, \tag{3.46}$$

$$\text{s.t. } \mathbf{x}^\intercal \mathbf{P}_{i,r_j}\mathbf{x} = 0 \;\forall\; i = 1,\ldots,N, \;\; j = 1,\ldots,6, \tag{3.47}$$

$$\mathbf{x}^\intercal \mathbf{P}_{i,c_j}\mathbf{x} = 0 \;\forall\; i = 1,\ldots,N, \;\; j = 1,\ldots,6, \tag{3.48}$$

$$\mathbf{x}^\intercal \mathbf{P}_{i,d_j}\mathbf{x} = 0 \;\forall\; i = 1,\ldots,N, \;\; j = 1,\ldots,9, \tag{3.49}$$

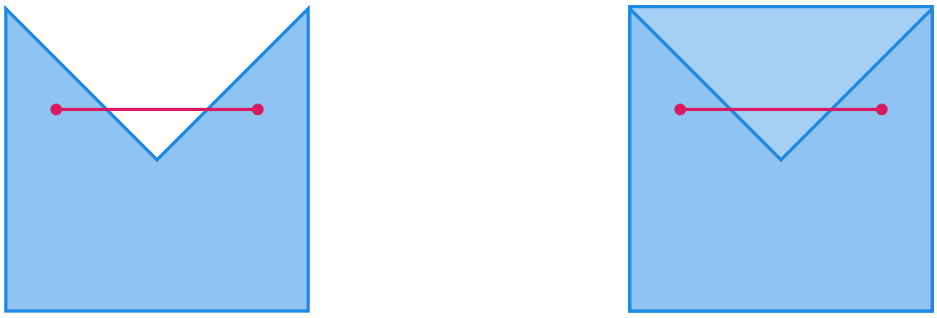
$$\mathbf{x}^\intercal \mathbf{P}_s\mathbf{x} = 1, \tag{3.50}$$

where each $\mathbf{P}_{i,r_j}, \mathbf{P}_{i,c_j}, \mathbf{P}_{i,d_j} \succeq 0$ *and* $\mathbf{P}_s \succeq 0$.

In Problem 4, Equations (3.47) to (3.50) enforce the row-orthogonality, column-orthogonality, cross-product, and homogenization constraints, respectively. For Chapters 7 and 8, we use a linear combination of the constraint matrices in Problem 4, which we also motivate later in this chapter. Due to the structure of problems in Chapters 7 and 8, the cost functions of our QCQPs satisfy $\mathbf{Q} \succeq 0$, but this relationship is not a requirement for an optimization problem to be a QCQP.

3.2.2 Convex Optimization

Problem 4 can now be solved using a convex relaxation. In this subsection, we review convex sets, functions, and problems to discuss the advantages of this relaxation.



(a) A nonconvex set.

(b) A convex set; more specifically, the convex hull of the nonconvex set in (a).

Figure 3.1: A visualization of nonconvex and convex sets.

Convex Sets

Let $\mathcal{Q} \subset \mathbb{R}^N$ be a set. If, for every pair of $\mathbf{x}_0, \mathbf{x}_1 \in \mathcal{Q}$, the line segment connecting $\mathbf{x}_0$ and $\mathbf{x}_1$ is a subset of $\mathcal{Q}$, then the set $\mathcal{Q}$ is said to be *convex*, that is,

$$\mathcal{Q} \text{ is convex} \iff \theta\mathbf{x}_0 + (1-\theta)\mathbf{x}_1 \in \mathcal{Q} \; \forall \; 0 \leq \theta \leq 1. \tag{3.51}$$

Figure 3.1 shows examples of nonconvex and convex sets. In this thesis, we enable convex relaxations for a subset of calibration problems by leveraging *convex cones*. A *cone* $\mathcal{Q}$ is a set, where, for every point $\mathbf{x} \in \mathcal{Q}$, the point $\theta\mathbf{x}$, such that $\theta \geq 0$, is also in the set. A *convex cone* $\mathcal{Q}$ must satisfy both the cone and convexity definitions. The two important convex cones for this thesis are the set of *positive semidefinite cones*, $\mathcal{Q} = \{\mathbf{X} \in \mathbb{R}^{N\times N} | \mathbf{X} \succeq 0\}$, and *positive definite cones* $\mathcal{Q} = \{\mathbf{X} \in \mathbb{R}^{N\times N} | \mathbf{X} \succ 0\}$.

Convex Functions

Consider the function $J : \mathbb{R}^N \mapsto \mathbb{R}$; the function J is a *convex function* if

1. $\text{dom}(J)$ is a convex set, and
2. $J(\theta\mathbf{x}_0 + (1-\theta)\mathbf{x}_1) \leq \theta J(\mathbf{x}_0) + (1-\theta)J(\mathbf{x}_1) \; \forall \; 0 \leq \theta \leq 1, \; \mathbf{x}_0, \mathbf{x}_1 \in \text{dom}(J)$.

To improve the interpretability of the second requirement, let the *epigraph* of a function be

$$\text{epi}(J(\mathbf{x})) = \{(\mathbf{x}, t) \mid \mathbf{x} \in \text{dom}(J), t \geq J(\mathbf{x})\}. \tag{3.52}$$

If $J(\mathbf{x})$ is convex, then the $\text{epi}(J(\mathbf{x}))$ is a convex set. Figure 3.2 shows an example of a convex function and the associated epigraph. Finally, if $-J(\mathbf{x})$ is convex, then $J(\mathbf{x})$ is *concave*.

Convex Optimization Problems

Using the definitions of convex sets and functions, we can define a general convex optimization problem.

Problem 5. *Convex Optimization Problem*

$$\min_{\mathbf{x}} J(\mathbf{x}), \tag{3.53}$$

$$s.t.\ f_i(\mathbf{x}) \leq 0 \ \forall\ i = 1, \ldots, N, \tag{3.54}$$

$$\mathbf{a}_i^\intercal \mathbf{x} + b_i = 0 \ \forall\ i = 1, \ldots, M, \tag{3.55}$$

where $J(\mathbf{x}), f_1(\mathbf{x}), \ldots, f_M(\mathbf{x})$ *are convex functions.*

Convex optimization formulations are advantageous because any local minimum is guaranteed to be a global minimum, which enables us to use simple gradient descent methods to find the global minimum. When solving nonconvex problems using first-order methods, such as Gauss-Newton, this guarantee does not exist. There are many methods for solving convex optimization problems (see Boyd and Vandenberghe [2004]), but, in this thesis, we treat these solvers as black boxes.

Given the definition of convex optimization problems, we can ask whether Problem 4 is a convex optimization problem. Since $\mathbf{Q} \succeq 0$, the cost function is clearly convex. However, the quadratic equality constraints are not affine, so the QCQP requires further manipulation before it is a convex problem.

3.2.3 Lagrangian Duality

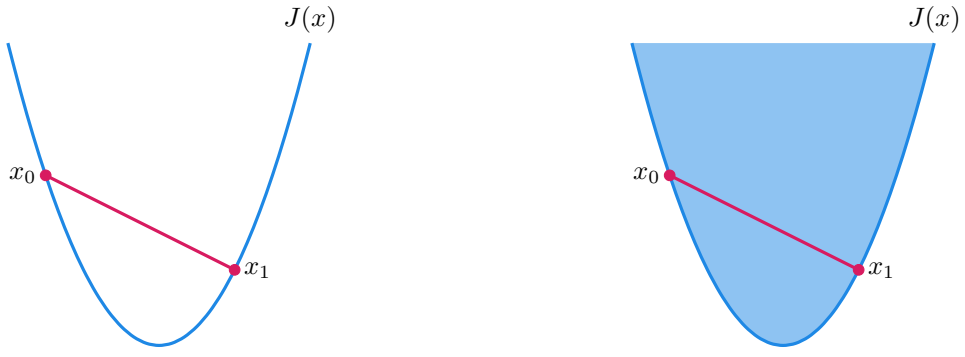
Every optimization problem has a *Lagrangian*. Given an arbitrary optimization problem with cost function, $J(\mathbf{x})$, inequality constraints, $f_i(\mathbf{x}) \leq 0 \ \forall\ i = 1, \ldots, N$, and equality constraints, $g_i(\mathbf{x}) = 0 \ \forall\ i = 1, \ldots, M$, the corresponding Lagrangian is

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}) = J(\mathbf{x}) + \sum_{i=1}^{N} \nu_i f_i(\mathbf{x}) + \sum_{i=1}^{M} \lambda_i g_i(\mathbf{x}), \tag{3.56}$$

where $\boldsymbol{\lambda}$ and $\boldsymbol{\nu}$ are the *Lagrangian multipliers* or the *dual variables*. These parameters can be interpreted as penalization terms for violating the inequality and equality constraints. Having defined the Lagrangian of an optimization problem, the *Lagrangian dual function* is

$$J_d(\boldsymbol{\lambda}, \boldsymbol{\nu}) = \inf_{\mathbf{x} \in \mathrm{dom}(f)} \mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}) = \inf_{\mathbf{x} \in \mathrm{dom}(J)} \left(J(\mathbf{x}) + \sum_{i=1}^{N} \nu_i f_i(\mathbf{x}) + \sum_{i=1}^{M} \lambda_i g_i(\mathbf{x}) \right). \tag{3.57}$$

Consequently, the Lagrangian dual function is a concave, affine function that serves as a lower bound on $p^* = J(\mathbf{x}^*)$, where $\mathbf{x}^*$ is the optimal value within the domain of the constraints if $\boldsymbol{\lambda} \geq 0$. The best, or maximal, lower



(a) A convex function $J(x)$.

(b) Epigraph of $J(x)$.

Figure 3.2: A visualization of the convex function $J(x)$ and the epigraph of $J(x)$.

bound of p^* provided by the dual function is $d^\star$, where $d^\star$ is the maximum of $J_d(\boldsymbol{\lambda}, \boldsymbol{\nu})$ with respect to the dual variables such that $\boldsymbol{\lambda} \geq 0$.

For all optimization problems, the inequality $d^* \leq p^*$ holds true and this property is called *weak duality*. However, for some optimization problems, a zero *duality gap* exists, that is $d^* = p^*$. In these cases, we say that *strong duality* holds. Optimization problems that demonstrate strong duality are particularly interesting because the solution of the simpler Lagrangian dual problem, $\boldsymbol{\lambda}^*$ and $\boldsymbol{\nu}^*$, determines the solution, $\mathbf{x}^*$, to the original nonconvex problem. One simple case where strong duality holds is for problems that satisfy *Slater's conditions*:

1. The optimization problem is a convex optimization problem.
2. There exists a feasible point $\mathbf{x}^*$, such that

$$f_i(\mathbf{x}^*) < 0 \; \forall \; i = 1, \ldots, N, \tag{3.58}$$

$$\mathbf{a}_i^\mathsf{T}\mathbf{x}^* + b_i = 0 \; \forall \; i = 1, \ldots, M. \tag{3.59}$$

Since Problem 4 is not a convex problem, Slater's conditions do not hold. Therefore, proving that strong duality holds for QCQPs requires other conditions. While Cifuentes et al. [2021] and Giamou [2023] present such conditions, this thesis primarily focuses on empirical strong duality. To clarify the concept of empirical strong duality, let $p(\mathbf{x})$ be the cost of an arbitrary optimization problem and $d(\boldsymbol{\lambda})$ be the cost of the associated Lagrangian dual. For a given dataset, if we find a primal-dual pair $(\mathbf{x}^\star, \boldsymbol{\lambda}^\star)$ such that $\boldsymbol{\lambda}^\star$ maximizes the Lagrangian dual and $p(\mathbf{x}^\star) = d(\boldsymbol{\lambda}^\star)$ is empirically observed, then we have found the global minimum because $p(\mathbf{x})$ cannot be less than $d(\boldsymbol{\lambda})$.[1]

3.2.4 Semidefinite Relaxations

This subsection presents the application of the concepts of convex problems and Lagrangian duality to QCQPs. Initially, we review semidefinite programs (SDPs), which are convex optimization problems. Then, we derive the semidefinite relaxation of a QCQP. Finally, we demonstrate that the Lagrangian dual of a QCQP is an SDP that is related to the semidefinite relaxation.

Semidefinite Programs

Semidefinite programs take two forms. The standard form is detailed below.

Problem 6. *Standard Formulation of a Semidefinite Program*

$$\min_{\mathbf{X} \in \mathbb{S}^n} \; \operatorname{tr}\left(\mathbf{C}\mathbf{X}\right), \tag{3.60}$$

$$s.t. \; \operatorname{tr}\left(\mathbf{A}_i\mathbf{X}\right) = b_i \; \forall i = 1, \ldots, N, \tag{3.61}$$

$$\mathbf{X} \succeq 0, \tag{3.62}$$

where $\mathbf{C}, \mathbf{A}_1, \ldots, \mathbf{A}_N \in \mathbb{S}^n$.

Another form of SDP is the following.

[1] Interested readers are referred to Giamou [2023] for the conditions pertinent to the QCQPs in this thesis.

Problem 7. *Semidefinite Program*

$$\min_{\mathbf{x}\in\mathbb{R}^n} \ \mathbf{c}^\intercal \mathbf{x}, \tag{3.63}$$

$$s.t. \ x_0\mathbf{F}_0 + x_1\mathbf{F}_1 + \cdots + x_N\mathbf{F}_N + \mathbf{G} \preceq 0, \tag{3.64}$$

$$\mathbf{A}\mathbf{x} = \mathbf{b}, \tag{3.65}$$

where $\mathbf{F}_0, \ldots, \mathbf{F}_N, \mathbf{G} \in \mathbb{S}^n$, *and* $\mathbf{A} \in \mathbb{R}^{p\times n}$.

Interestingly, with some algebraic manipulation, the standard form SDP is the Lagrangian dual of Problem 7. In fact, these problems demonstrate strong duality: both forms of the SDPs return the same solution.

Semidefinite Relaxation of a QCQP

The semidefinite relaxation of Problem 4 results in an optimization problem that takes the form of Problem 6.

Problem 8. *Semidefinite Relaxation of Problem 4*

$$\min_{\mathbf{x}} \ \mathrm{tr}\left(\mathbf{Q}\mathbf{X}\right), \tag{3.66}$$

$$s.t. \ \mathrm{tr}\left(\mathbf{P}_{i,r_j}\mathbf{X}\right) = 0 \ \forall \ i = 1, \ldots, N, \ j = 1, \ldots, 6, \tag{3.67}$$

$$\mathrm{tr}\left(\mathbf{P}_{i,c_j}\mathbf{X}\right) = 0 \ \forall \ i = 1, \ldots, N, \ j = 1, \ldots, 6, \tag{3.68}$$

$$\mathrm{tr}\left(\mathbf{P}_{i,d_j}\mathbf{X}\right) = 0 \ \forall \ i = 1, \ldots, N, \ j = 1, \ldots, 9, \tag{3.69}$$

$$\mathrm{tr}\left(\mathbf{P}_s\mathbf{X}\right) = 1, \tag{3.70}$$

where $\mathbf{X} = \mathbf{x}\mathbf{x}^\intercal$.

Importantly, we drop or *relax* the nonconvex constraint that $\mathrm{rank}(\mathbf{X}) = 1$. While we could use this relaxation for our problems in Chapters 7 and 8, we cannot solve problems of this form using the fast solver described in Garstka et al. [2021].

Lagrangian Dual of a QCQP

Since the semidefinite relaxation of Problem 4 takes the form of a standard semidefinite program, the Lagrangian of Problem 4 must also be a semidefinite program. Given Problem 4, the Lagrangian of the problem is

$$\mathcal{L}(\mathbf{x}, \nu) = \mathbf{x}^\intercal \mathbf{Z}(\boldsymbol{\nu})\mathbf{x} + \nu_N, \tag{3.71}$$

$$\mathbf{Z}(\boldsymbol{\nu}) = \mathbf{Q} + \sum_{i=0}^{N}\sum_{j=1}^{6} \nu_{i,r_j}\mathbf{P}_{i,r_j} + \sum_{i=0}^{N}\sum_{j=1}^{6} \nu_{i,c_j}\mathbf{P}_{i,c_j} + \sum_{i=0}^{N}\sum_{j=1}^{9} \nu_{i,d_j}\mathbf{P}_{i,d_j} + \nu_s\mathbf{P}_s, \tag{3.72}$$

where $\boldsymbol{\nu}$ is the concatenation of all 21N + 1 dual variables. Equation (3.72) is the linear combination of the rotation constraint matrices that we discussed when formulating the QCQP. Since $\mathbf{Z}(\boldsymbol{\nu})$ depends on $\mathbf{Q}$, the structure of $\mathbf{Z}(\boldsymbol{\nu})$ is dependent on $\mathbf{x}$, which is problem specific. Consequently, we leave the explicit definitions of the $\mathbf{Z}(\boldsymbol{\nu})$ matrices to Chapters 7 and 8. In those chapters, we highlight the form of the rotation matrix constraints in $\mathbf{Z}(\boldsymbol{\nu})$. While other calibration problems may include inequality constraints, the problems in Chapters 7 and 8 consist of

only equality constraints. Consequently, Equation (3.71) has no dual parameters for the inequality constraints. Minimizing $\mathcal{L}(\mathbf{x}, \boldsymbol{\nu})$ with respect to $\mathbf{x}$ reveals that $\mathbf{Z}(\boldsymbol{\nu}) \succeq 0$ in order to have a defined dual problem; otherwise, the optimal cost is negative infinity. As a result, the dual problem is as follows.

Problem 9. *Lagrangian Dual of Problem 4*

$$\max_{\boldsymbol{\nu}} \nu_N, \tag{3.73}$$

$$s.t.\ \mathbf{Z}(\nu) \succeq 0, \tag{3.74}$$

which is the same form as Problem 7. Importantly, both Problem 8 and Problem 9 produce the same solution, so either method can be used to solve the original QCQP. In this thesis, we solve the Lagrangian dual of our QCQPs. However, we refer to the method as semidefinite relaxation because, as discussed previously, the two problems result in the same solution.

3.3 Summary

In this chapter, we reviewed batch nonlinear optimization and convex optimization problems. We discussed an on-manifold Gauss-Newton approach to solving nonlinear optimization problems. After presenting the Gauss-Newton method, we cast a subset of nonlinear calibration problems as quadratically constrained quadratic programs (QCQPs). We defined convex sets, functions, and optimization problems. We showed that the Lagrangian dual is the lower bound of an arbitrary optimization problem. Finally, we discussed the semidefinite relaxation of QCQPs. The relaxed problem is convex, so solvers for standard semidefinite programs (SDPs) find the global optimum. If the final values of the QCQP and SDP cost functions are equal at the global optimum, then we have found the optimal solution to the QCQP.

Chapter 4

Observability and Identifiability

In Chapter 2, we informally asserted that calibration problems are only solvable with a sufficient number of measurements and with sufficient platform excitation. This chapter explores concepts that are required to prove these assertions, which are essential for accurate calibration of real world systems. In Section 4.1, we define system representations of optimization problems, indistinguishability, u-indistinguishability, observability, and identifiability. Sections 4.2 and 4.3 discuss criteria to prove observability and identifiability. These criteria enable us to determine when the system is *degenerate* (i.e., the combination of states and controls where the system loses observability or identifiability). In turn, we will be able to establish the excitation requirements of the system that determine whether a calibration problem is well posed.

4.1 Definitions

In this thesis, we describe optimization problems as systems of the form

$$S \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}\left(\mathbf{x}(t), \mathbf{u}(t)\right), \\ \mathbf{y}(t) = \mathbf{h}\left(\mathbf{x}(t)\right), \end{cases} \tag{4.1}$$

where $\mathbf{x} \in \mathcal{M}$, a connected C^∞ manifold of dimension m. In this definition, the function $\mathbf{f}(\cdot)$ is the continuous-time motion model, which we can use to derive the discrete motion model in Equation (3.1). Further, the continuous-time motion and measurement models of the system are noiseless unlike our optimization problems.

Having defined our system, we can discuss the two forms of observability pertinent to this thesis. Hermann and Krener [1977] define observability in relation to the notion of indistinguishablility.

Definition 1. *Let $\mathbf{x}_0$ and $\mathbf{x}_1$ be two states of the system S. These two states are indistinguishable if they realize the same input-output map for every admissible input.*

They then define observability as follows.

Definition 2. *A system S is observable at $\mathbf{x}$ if the set of states that are indistinguishable from $\mathbf{x}$ is singleton. A system S is observable if it is observable for every $\mathbf{x} \in \mathcal{M}$.*

Observability is a global property that can be difficult to prove for many systems, as we discuss in Section 4.2.2. One way forward is to consider a more restricted problem. Hermann and Krener [1977] define this problem in relation to the notion of U-indistinguishability.

Definition 3. *Let $\mathcal{U}$ be a subset of $\mathcal{M}$ with $\mathbf{x}_0, \mathbf{x}_1 \in \mathcal{U}$. We say $\mathbf{x}_0$ is U-indistinguishable from $\mathbf{x}_1$ if for every control $\mathbf{u}(t)$, whose trajectories from $\mathbf{x}_0$ and $\mathbf{x}_1$ both lie in $\mathcal{U}$, fails to distinguish between $\mathbf{x}_0$ and $\mathbf{x}_1$. That is, if $\mathbf{x}_0(t), \mathbf{x}_1(t) \in \mathcal{U}$ for $t \in [t_0, t_1]$, then $\mathbf{h}(\mathbf{x}_0(t)) = \mathbf{h}(\mathbf{x}_1(t))$ for every control $\mathbf{u}(t)$.*

Making use of U-indistinguishability, Hermann and Krener [1977] define locally weak observability as shown below.

Definition 4. *A system is locally weakly observable at $\mathbf{x}$ if there exists an open neighbourhood V on $\mathcal{M}$, such that, for each open neighbourhood U contained in V, the set of U-indistinguishable states is a singleton. A system is locally weakly observable if the system is locally weakly observable at every $\mathbf{x}$.*

The benefit of the local weak observability criterion is that it lends itself to a relatively straightforward algebraic test, which we discuss in Section 4.2.2. Often, we prove that a system is locally weakly observable at a generic $\mathbf{x}$, but this does not guarantee locally weak observability for all controls $\mathbf{u}(t)$. When there is insufficient excitation to render the system locally weakly observable, we say the system is degenerate.

To provide a trivial example of a system that is locally weakly observable at $\mathbf{x}$, but not observable, consider a system where $\mathbf{y} = \mathrm{vec}\,(\mathbf{R}(\mathbf{q}))$, $\mathbf{R} \in \mathrm{SO}(3)$, $\mathbf{x} = \mathbf{q}$, and $\mathbf{q} \in \mathrm{S}^3$. We can use the mapping from $\mathrm{SO}(3) \to \mathrm{S}^3$ to recover a quaternion that produces $\mathbf{y}$. We recall, from Section 2.1.2, that this mapping has two indistinguishable antipodal solutions, so the system cannot be observable by definition. However, we can choose an open neighbourhood on the manifold of S^3 that contains only one solution, so the system is locally-weakly observable at $\mathbf{x}$.

Unlike observability criteria, identifiability criteria are not instantaneous because they use multiple measurement times to prove that a solution is isolated. Anstett-Collin et al. [2020] define identifiability relative to the notion of U-indistinguishability.

Definition 5. *Given the input-output behaviour of a system (i.e., $\{\mathbf{y}(\mathbf{x}(t_0)), \ldots, \mathbf{y}(\mathbf{x}(t_0), \mathbf{u}(t_0), \ldots, \mathbf{u}(t_k))\}$), a system is identifiable at $\mathbf{x}(t_0)$ if there exists an open neighbourhood U such that the set of U-indistinguishable states is a singleton.*

Consequently, proving that a system is observable implies identifiability, but identifiability is often easier to prove than observability. Finally, as shown in Anguelova [2007] and Kelly et al. [2021], some systems, such as time-delay systems, may only be identifiable and not observable.

4.2 Observability

In this section, we review methods to determine the observability of linear, nonlinear, and time-delayed nonlinear systems. In Section 4.2.1, we discuss a criterion to determine the observability of linear systems. Section 4.2.2 presents a criterion to prove that a nonlinear system is locally weakly observable.

4.2.1 Linear Systems

For linear observability analyses, we consider noiseless linear systems that take the form

$$S \begin{cases} \dot{\mathbf{x}}(t_i) = \mathbf{A}(t_i)\mathbf{x}(t_i) + \mathbf{B}(t_i)\mathbf{u}(t_i), \\ \mathbf{y}(t_i) = \mathbf{C}(t_i)\mathbf{x}(t_i), \end{cases} \tag{4.2}$$

where $\mathbf{A}(t) : \mathbb{R} \to \mathbb{R}^{N\times N}$, $\mathbf{B}(t) : \mathbb{R} \to \mathbb{R}^{N\times P}$, $\mathbf{C}(t) : \mathbb{R} \to \mathbb{R}^{M\times N}$, are, respectively, the motion, control and measurement matrices. Given the discrete measurements at times t_0 to t_k, Barfoot [2024] shows that the state of S at t_0 to t_k can be uniquely recovered if

$$\mathcal{O} = \begin{bmatrix} \mathbf{C}_0 \\ \mathbf{C}_1\mathbf{A}_0 \\ \vdots \\ \mathbf{C}_k \prod_{i=k-1}^{0} \mathbf{A}_i \end{bmatrix} \tag{4.3}$$

is full column rank, where $\mathbf{A}_k = \mathbf{A}(t_k)$ and $\mathbf{C}_k = \mathbf{C}(t_k)$. If the matrix $\mathcal{O}$ is full column rank, then the system is observable. When $\mathcal{O}$ is not full-column rank, the set of indistinguishable states for the system is not a singleton and we say that the system is unobservable.

Following Barfoot [2024], a time-invariant linear system (i.e., $\mathbf{A}_k = \mathbf{A}$ and $\mathbf{C}_k = \mathbf{C}$) can be simplified using the Cayley-Hamilton theorem. Specifically, since $\mathbf{A}$ is a $N \times N$ matrix, any power of $\mathbf{A}$ greater than or equal to N is a linear combination of the powers of $\mathbf{A}$ less than N. Therefore, if the system is time-invariant, then the rank of $\mathcal{O}$ in Equation (4.3) is equivalent to the rank of

$$\mathcal{O} = \begin{bmatrix} \mathbf{C} \\ \mathbf{C}\mathbf{A} \\ \vdots \\ \mathbf{C}\mathbf{A}^{N-1} \end{bmatrix}. \tag{4.4}$$

Importantly, we only need to propagate the state forward $N - 1$ timesteps to determine if the set of indistinguishable states for a time-invariant linear system is a singleton.

4.2.2 Nonlinear Systems

Similar to linear systems, we want to determine if any solution, $\mathbf{x} \in \mathcal{M}$, to a nonlinear system is observable at a point on the manifold, $\mathcal{M}$. Since nonlinear systems do not have linear measurement or motion models, we cannot use the Cayley-Hamilton theorem. Determining the observability of a nonlinear system may involve travelling for a long distance or period of time on $\mathcal{M}$. In this case, to demonstrate that nonlinear systems are locally weakly observable, we use Hermann and Krener's [1977] criterion, where we analyze systems in control-affine form,

$$S \begin{cases} \dot{\mathbf{x}} = \mathbf{f}_0(\mathbf{x}) + \sum_{j=1}^{p} \mathbf{f}_j(\mathbf{x})u_j \\ \mathbf{y} = \mathbf{h}(\mathbf{x}) \end{cases}, \tag{4.5}$$

with the drift vector field $\mathbf{f}_0(\mathbf{x})$ and control inputs u_j (for $j = 1, \ldots, p$). The Lie derivative, or directional derivative, of a smooth scalar function $\mathbf{h}$ with respect to a smooth vector field $\mathbf{f}$ at the point $\mathbf{x}$ is

$$L_{\mathbf{f}}\mathbf{h}(\mathbf{x}) = \frac{\partial \mathbf{h}(\mathbf{x})}{\partial \mathbf{x}}\mathbf{f}(\mathbf{x}). \tag{4.6}$$

The n^{th} Lie derivative of $\mathbf{h}$ with respect to $\mathbf{x}$ along $\mathbf{f}$ is defined recursively as

$$L^n_{\mathbf{f}}\mathbf{h}(\mathbf{x}) = \frac{\partial L^{n-1}_{\mathbf{f}}\mathbf{h}(\mathbf{x})}{\partial \mathbf{x}}\mathbf{f}(\mathbf{x}), \tag{4.7}$$

where $L^0\mathbf{h}(\mathbf{x}) = \mathbf{h}(\mathbf{x})$.

As shown in Hermann and Krener [1977], if, for a given state $\mathbf{x}$, the matrix

$$\mathcal{O} = \begin{bmatrix} \nabla_{\mathbf{x}} L^0\mathbf{h}(\mathbf{x}) \\ \nabla_{\mathbf{x}} L^1\mathbf{h}(\mathbf{x}) \\ \vdots \end{bmatrix} \tag{4.8}$$

has full column rank, then the system is locally weakly observable. We note that the matrix $\mathcal{O}$ has potentially an infinite number of rows, but it is sufficient to show that a finite number of rows yield a matrix of full column rank. Additionally, once $\nabla_{\mathbf{x}} L^i_{\mathbf{f}}\mathbf{h}$ produces a linearly dependent row, subsequent Lie derivatives also produce them. The matrix $\mathcal{O}$ can become degenerate for certain combinations of states and controls, which establishes the excitation requirements of the system. For some systems, the rank proof is complex, and leveraging abstracted sensor representations may simplify the proof. For example, Kelly and Sukhatme [2011] use the scaled pose representation of a monocular camera to prove the locally weak observability of a camera-IMU system. Finally, in Section 5.4, we leverage this proof to demonstrate that our radar-camera extrinsic calibration problem is locally weakly observable.

4.3 Identifiability

In this subsection, we present two criteria to help investigate the identifiability of nonlinear systems. As shown in Chapter 3, many of our estimation problems involve minimizing the square of a nonlinear error that is defined by our system. We review a nonlinear least-squares identifiability criterion in Section 4.3.1 and then show, in Section 4.3.2, we show how stacking the nonlinear observability matrix from Hermann and Krener [1977], at different times, can demonstrate the identifiability of certain systems. This stacked identifiability matrix criterion is easier to evaluate than the nonlinear least-squares criterion.

4.3.1 Nonlinear Least-Squares Systems

We can determine the identifiability of Equation (4.1) using the criterion from Jacquez [1991], which is

$$\mathcal{O} = \begin{bmatrix} \nabla_{\boldsymbol{\theta}} \mathbf{h}(\boldsymbol{\theta}, \mathbf{x}(t_0)) \\ \nabla_{\boldsymbol{\theta}} \mathbf{h}(\boldsymbol{\theta}, \mathbf{x}(t_1)) \\ \vdots \\ \nabla_{\boldsymbol{\theta}} \mathbf{h}(\boldsymbol{\theta}, \mathbf{x}(t_N)) \end{bmatrix}, \tag{4.9}$$

where $\boldsymbol{\theta}$ is a set of time-invariant states. If $\mathcal{O}$ is full column rank, then the system is identifiable. Critically, to compute the matrix $\mathcal{O}$, we must solve the differential equations that govern the time-varying states. As in Section 4.2.2, the matrix $\mathcal{O}$ can lose full column rank under certain controls. This criterion is not practical for all nonlinear systems because there might not be a closed-form solution that propagates the state from t_i to t_j.

4.3.2 Stacked Nonlinear Observability

Here, we leverage abstracted sensor representations to sidestep solving differential equations in nonlinear identifiability analyses. As shown in Kelly and Sukhatme [2011], Hewitt and Marshall [2015], and Li and Mourikis [2014], we can use the abstracted sensor representations in Chapter 2 to simplify the application of the criterion from Hermann and Krener [1977]. By extending this notion, we can simplify some systems, such that

$$S \begin{cases} \dot{\mathbf{x}} = \mathbf{0}, \\ \mathbf{y} = \mathbf{h}(\mathbf{x}, t). \end{cases} \tag{4.10}$$

The identifiability criterion for this system is

$$\mathcal{O} = \begin{bmatrix} \nabla_{\mathbf{x}} \mathbf{h}(\mathbf{x}, t_0) \\ \nabla_{\mathbf{x}} \mathbf{h}(\mathbf{x}, t_1) \\ \vdots \end{bmatrix}, \tag{4.11}$$

which we refer to as stacked nonlinear observability. If $\mathcal{O}$ is full column rank, then the system is identifiable. In this criterion, the matrix $\mathcal{O}$ may contain components of the output $\mathbf{y}$ and certain sets of measurements may cause the system to become degenerate. While eliminating the time varying states simplifies determining identifiablity and excitation requirements, this analysis ignores the vector fields and subsequent differential equations that govern the time-varying states, which may obscure certain rare degenerate systems.

4.4 Summary

In this chapter, we reviewed observability and identifiability. Defining indistinguishability and U-indistinguishability enabled us to define various types of linear and nonlinear observability and identifiabilty. Having defined observability, we presented the criteria for linear and nonlinear systems.Then, we presented two nonlinear identifiability criteria. The first criterion requires solving the differential equations that govern the states, which we are able to avoid using the second criterion.

Chapter 5

Spatiotemporal Calibration of 3D Radar-Camera Pairs

With the background material in place, we now present the first contribution of the thesis, a targetless spatiotemporal calibration algorithm for radar-camera pairs. Prior methods, such as that in Peršić et al. [2021b], are reprojection-based calibration schemes that require cross-modal data association. To simplify the data association problem, these methods leverage specialized targets (i.e., radar trihedral retroreflectors and visual fiducial markers). Consequently, the sensors must have overlapping fields of view and the environment must be instrumented, which limits possible sensor configurations and potential calibration venues.

We provide the first method for estimating the spatiotemporal calibration parameters between a 3D mm-wave radar and monocular camera *without* the use of specialized targets. In turn, we enable spatiotemporal calibration of monocular camera-3D radar pairs in arbitrary configurations and in environments without any radar retroreflectors or visual fiducials. Prior to describing our approach, we begin in Section 5.1 by surveying extrinsic and spatiotemporal calibration techniques for sensor pairs where one (or both) of the sensors is a mm-wavelength radar. Then, in Section 5.2, we briefly review the process of radar ego-velocity estimation. Section 5.3 discusses our novel spatiotemporal radar-camera calibration method, initially introduced in Wise et al. [2021] and extended in Wise et al. [2023]. We examine the observability and identifiability of the extrinsic and spatiotemporal calibration problems in Sections 5.4 and 5.5, respectively. In Section 5.6, we demonstrate the accuracy and flexibility of our algorithm by reporting on a variety of simulation studies and real-world experiments. In our simulation studies, we analyze the accuracy of spatiotemporal calibration with varying amounts of sensor noise. Our real-world experiments show that our algorithm is able to match the accuracy of an existing, target-based method and that we are able to perform calibration in varied environments, including on board an AV. Finally, Section 5.7 summarizes our work on spatiotemporal calibration of 3D radar-camera pairs and outlines future research directions.

5.1 Related Work

In this section, we discuss pairwise extrinsic and spatiotemporal calibration methods that involve at least one mm-wavelength radar. Each subsection focuses on the calibration methods that use a particular measurement error model defined in Section 2.4. To start, Section 5.1.1 reviews homography-based calibration schemes for 2D radars. Sections 5.1.2 and 5.1.3 detail target-based and target-free extrinsic calibration algorithms, respectively,

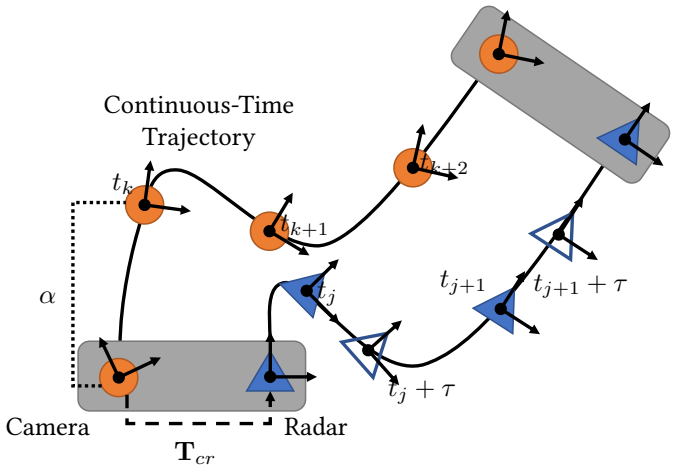


Figure 5.1: The radar (triangle) and camera (circle) are assumed to be rigidly connected. Our calibration problem involves estimating the transform between the camera and radar, $\mathbf{T}_{cr}$, the translation scale factor, α, for the camera pose measurements, and the temporal offset, τ. The unfilled triangles represent radar measurements at 'shifted' points in time due to the offset bias. The radar ego-velocity estimate will not be correct unless this bias is considered.

that rely on feature detection and matching. In Section 5.1.4, we review extrinsic calibration methods that relate the instantaneous radar ego-velocity to the motion of the second sensor. Section 5.1.5 surveys reprojection-based spatiotemporal calibration. We summarize the limitations of current mm-wave radar calibration algorithms relative to our own in Table 5.1.

5.1.1 Target-Based Homographic Error Methods

Since 2D radars are incapable of directly measuring the elevation of a target, early-stage extrinsic calibration algorithms assume that the radar measurement is purely planar and perform homography-based calibration. We can further subdivide these early algorithms by the homography matrix parameterization, either an affine approximation or the full projective transformation. The method of Kim and Jeon [2014] solves for the affine homography matrix. In contrast to Kim and Jeon [2014], the methods of Sugimoto et al. [2004], Wang et al. [2011] and Kim et al. [2018] solve for the full projective homography matrix. While these methods can successfully calibrate radar-camera pairs, the radar sensors do often detect off-plane targets, so an arbitrary radar-camera measurement pair is populated with outliers.

To improve the robustness of homography-based radar-camera calibration algorithms to off-plane targets, all of the methods mentioned in this subsection rely on trihedral retroreflector targets and fiducial camera targets. Use of these specialized targets confers three major benefits. First, Sugimoto et al. [2004] note that 2D radar units typically measure a maximum radar cross section (RCS) when a retroreflector lies on the plane of zero elevation in the radar reference frame; the return intensity decreases for reflectors that lie above or below this plane. By filtering the returns using RCS intensity, Sugimoto et al. [2004] ensure that the radar-camera measurement pairs are in the radar plane of zero elevation. Second, the trihedral retroreflector targets produce point-like reflections. Finally, the targets greatly simplify the radar-camera data association or correspondence problem.

While targets enable reliable and accurate homography matrix estimation, in-situ homography-based calibration is impractical outside of laboratory settings. Trihedral reflector targets do not occur in the real world, so these targets become an infrastructure requirement. Even if targets are available, every system must have all radars configured at the same height or the targets will be off-plane. If we avoid using retroreflectors, then de-

Table 5.1: Summary of sensor-to-sensor calibration methods where one sensor in the pair is a mm-wavelength radar. We include our work from Cheng et al. [2023], which we present in Chapter 6.

Method	Published Works	Targetless	Non-Overlapping Fields of View	No Additional Sensing Modalities	No Hand Measurements	Temporal Offset Estimation
Homography-Based Extrinsic Calibration	Sugimoto et al. [2004] Wang et al. [2011] Kim and Jeon [2014] Kim et al. [2018]	✗	✗	✓	✓	✗
Target Reliant Reprojection-Based Extrinsic Calibration	El Natour et al. [2015] Kim et al. [2017] Domhof et al. [2019] Peršić et al. [2019] Olutomilayo et al. [2021]	✗	✗	✓	✓	✗
Target-Free Reprojection-Based Extrinsic Calibration	Schöller et al. [2019] Heng [2020] Peršić et al. [2021a] Burnett et al. [2023]	✓	✓	✗	✗	✗
Ego-Velocity-Based Extrinsic Calibration	Kellner et al. [2015] Doer and Trommer [2020a]	✓	✓	✓	✓	✗
Target-Reliant Reprojection-Based Spatiotemporal Calibration	Lee et al. [2020] Peršić et al. [2021b]	✗	✗	✓	✓	✓
Our Methods	Wise et al. [2021] Wise et al. [2023] Cheng et al. [2023]	✓	✓	✓	✓	✓

termining on- and off-plane measurements using RCS becomes a non-trivial task because the RCS of real-world targets, such as pedestrians or cars, varies based on material composition, position in the radar lobe, and relative orientation to the radar. Additionally, determining radar-camera measurement correspondences without the visual fiducial targets is very challenging.

5.1.2 Target-Based Reprojection Error Methods

The most common error metric for mm-wavelength radar extrinsic calibration is reprojection error (as described in Section 2.4). Some reprojection-based methods draw inspiration from the homography-based methods and retain the horizontal plane constraint (for specialized targets). However, one goal of reprojection-based methods is to eliminate the horizontal plane constraint. Even though reprojection-based algorithms successfully eliminate the horizontal plane constraint in many cases, the continued use of specialized targets limits system configurations and calibration venues. An empirical comparison of homography- and reprojection-based methods in Oh et al. [2018] suggests that both methods have approximately the same accuracy.

Similar to homography methods, there are reprojection-based methods that assume that the radar measurements originate from the horizontal plane. To perform radar-infrared (IR) camera extrinsic calibration, Kim et al. [2017] estimate the $\mathrm{SE}(3)$ transformation that aligns radar and IR camera measurements of a specialized target. Olutomilayo et al. [2021] estimate the $\mathrm{SE}(2)$ transformation that aligns the measured point clouds from two rigidly-connected 2D mm-wavelength radars. While both of these approaches can solve for the extrinsic calibration parameters, in-situ calibration is not feasible.

To relax the horizontal plane constraint, some reprojection-based algorithms account for the elevation ambiguity in the radar measurements. The approaches of El Natour et al. [2015], Domhof et al. [2019], and Peršić et al. [2019] treat all radar measurements as lying on spherical arcs with constant range and azimuth (i.e., the measurements vary only in elevation). To account for the elevations of the retroreflector targets relative to the horizontal radar sensing plane, these methods introduce additional calibration constraints by designing specific

target arrangements (El Natour et al. [2015], Domhof et al. [2019]) or by explicitly modelling the RCS of the target returns (Peršić et al. [2019]).

While eliminating the horizontal plane constraint is a key step towards in-situ calibration, reprojection-based methods have two requirements that make this infeasible in practice. First, the infrastructure requirements (i.e., specialized targets) prevent calibration in arbitrary environments. Second, all of the methods above are restricted to sensor pairs that share overlapping fields of view, which may be impossible to satisfy for certain sensor configurations.

5.1.3 Target-Free Reprojection-Error Methods

In contrast to methods that rely on specialized radar retroreflectors, target-free or 'targetless' algorithms estimate the radar-to-sensor transform by aligning identifiable environment features (observed by both sensors). These methods often use distance-based correspondences to identify and match features viewed by both sensors. Due to the high outlier rate and inconsistent detection of features in radar measurements, the methods frequently require a prior estimate of the calibration parameters, a large number of jointly viewed features, or a map of the environment.

Due to the challenge of consistently tracking environmental features across radar scans, some target-free methods require hand-measurement of the translation between the sensors and only estimate the relative rotation. Schöller et al. [2019] train a neural network to correct an inaccurate rotation estimate between a stationary, 2D radar-camera pair using radar detections of vehicles and raw camera images. Peršić et al. [2021a] align tracked features to determine the yaw angle between 2D radar-camera and radar-lidar pairs. Similar to the reprojection-based algorithms, these methods require overlapping fields of view. Unlike Schöller et al. [2019] and Peršić et al. [2021a], Burnett et al. [2023] estimate the transform between a 2D mm-wavelength radar-lidar pair using a reprojection-based method with no prior knowledge of the system. In particular, their method leverages scan matching to align radar and lidar point clouds, which requires a large number of jointly-observed features. The sensors considered in Burnett et al. [2023] have a 360° field of view; radars and lidars with narrow fields of view may not observe enough features to enable scan matching.

By constructing a lidar map using a known vehicle trajectory, Heng [2020] enables extrinsic calibration of 3D mm-wavelength radar-lidar pairs, where each sensor has a narrow field of view. In Heng [2020], the radar-lidar extrinsic calibration parameters are determined by minimizing point-to-plane and velocity residuals. The point-to-plane residual ensures that the extrinsic calibration parameters align each radar detection with a plane formed by a fixed number of closest points in the lidar map. A robust cost function minimizes the influence that outlier radar detections have on the estimated calibration parameters. By associating the radar detections with the lidar map, Heng [2020] assumes that each radar detection is a stationary (i.e., relative to the world frame) feature. While this approach can be used for radar-lidar pairs that do not have overlapping fields of view, the method requires the known trajectory of the vehicle and construction of a dense map.

5.1.4 Ego-Velocity Error Methods

Instead of using feature positions, a subset of extrinsic calibration algorithms fuse ego-velocity and ego-motion measurements from a radar and a second sensor, respectively. Since the motion of each sensor is estimated separately, these methods do not perform radar or cross-modal data association and are inherently 'target-free.' Kellner et al. [2015] estimate the rotation between a car-mounted 2D radar and an IMU by minimizing the difference in estimated lateral velocities, expressed in the radar frame. While the radar ego-velocity measurements

provide lateral velocity directly, determining the lateral velocity of the radar from IMU measurements requires the IMU rotational velocity and accurate knowledge of the radar-IMU translation. Doer and Trommer [2020a] extend the approach in Kellner et al. [2015] to estimate the full extrinsic calibration for a 3D radar-IMU pair. Using simulated radar ego-velocity estimates corrupted by zero-mean Gaussian noise, their method achieves a spatial calibration accuracy of 5 cm and 5°. Each these techniques rely on ad hoc temporal calibration schemes.

5.1.5 Target-Based Spatiotemporal Calibration

To date, two radar spatiotemporal calibration algorithms have appeared in the literature, by Lee et al. [2020] and by Peršić et al. [2021b]. The algorithm of Lee et al. [2020] first calibrates the 2D radar-lidar spatial transform using the method of Peršić et al. [2019]. As a second step, the lidar measurements are expressed in the radar reference frame and the azimuth error to distant targets is minimized to determine the temporal offset between the sensors. Finally, the algorithm iterates between the spatial and temporal calibration steps until convergence. Peršić et al. [2021b] represent the trajectory of a target moving through the fields of view of multiple sensors using a continuous-time Gaussian process model. This representation allows their algorithm to estimate the spatiotemporal calibration parameters by aligning the sensors' trajectories. Discussion in Rehder et al. [2016b] and in Peršić et al. [2021b] suggests that jointly estimating all parameters as part of one maximum likelihood estimation problem yields superior accuracy . Notably, since both methods rely on known targets, they have the same limitations as the methods discussed in Section 5.1.2.

5.2 Radar Ego-Velocity Estimation

Given the importance of radar ego-velocity estimation to this chapter, we present a brief discussion of a standard radar ego-velocity estimator. We break the discussion into three subtopics. Initially, we review the operating principles and measurement details for 2D and 3D radars, derive an ego-velocity estimator, and explain common outlier rejection schemes. We then examine the effects of multipath reflections and interference on radar range-rate measurements. Finally, we discuss how to estimate the linear and rotational velocity of sensor platform reference frames that are rigidly connected to the radar.

Essentials of Radar Ego-Velocity Estimation

Both 2D and 3D radar sensors operate on the same basic principle: an EM pulse is emitted from the radar antenna, reflects off radar-opaque targets in the environment, and returns to the sensor. These EM waves are

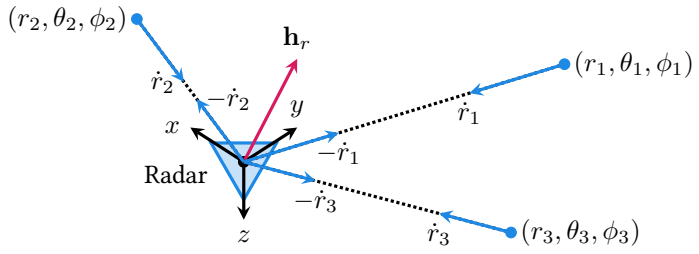


Figure 5.2: Illustration of the radar measurement model considered in this chapter. The radar electromagnetic (EM) wave reflects off of three (or more) non-collinear, stationary landmarks in the environment, yielding azimuth, elevation, and range-rate measurements to each landmark. From these data, we estimate the radar velocity relative to the world reference frame, expressed in the radar reference frame.

able to pass through rain, snow, and other obscurants. By measuring the time of flight and phase of the return pulse, 2D radars can determine the azimuth, θ_i, range, r_i, range-rate (velocity in the radial direction), $\dot{r}_i$, and cross-section (reflectivity) of multiple targets, $i = 1, \ldots, N$. In addition to all of the quantities measured by a 2D radar, a 3D radar can measure the elevation of a target, ϕ_i. Importantly, the azimuth and elevation measurements are relative to the radar boresight. The radar boresight is the vector that points in the direction of EM wave propagation and is coincident with the axis of symmetry for the EM pulse.

Let the static world and moving radar frames be $\underrightarrow{\mathcal{F}}_w$ and $\underrightarrow{\mathcal{F}}_r$, respectively. At each time t_j, the radar detects N static environmental features in $\underrightarrow{\mathcal{F}}_w$. The resulting sets of 3D radar measurements, expressed in $\underrightarrow{\mathcal{F}}_r$, are

$$\mathcal{R} = \left\{ \begin{bmatrix} r_1 & \theta_1 & \phi_1 & \dot{r}_1 \end{bmatrix}, \ldots, \begin{bmatrix} r_N & \theta_N & \phi_N & \dot{r}_N \end{bmatrix} \right\}. \tag{5.1}$$

Following Stahoviak [2019] and Doer and Trommer [2020a], the 3D range-rate measurement of feature i is

$$\dot{r}_i = -\begin{bmatrix} \sin(\theta_i)\cos(\phi_i) & \cos(\theta_i)\cos(\phi_i) & \sin(\phi_i) \end{bmatrix} \mathbf{h}_r^j, \tag{5.2}$$

where $\mathbf{h}_r^j \in \mathbb{R}^3$. The quantity $\mathbf{h}_r^j$ is the velocity of the radar relative to the world reference frame, expressed in the radar reference frame. We refer to $\mathbf{h}_r^j$ as either the radar ego-velocity. Importantly, for 2D radars, we assume that the feature lie on (and radar moves on) a horizontal plane, that is $\phi_i = 0 \ \forall \ i = 1, \ldots, N$ and $\mathbf{h}_r^j \in \mathbb{R}^2$. Equation (5.2) has a negative sign because we wish to estimate the velocity of the radar with respect to the static world frame, and not vice versa. Figure 5.2 depicts the relationship between the range-rate of stationary features and ego-velocity of the radar. Stacking N (negated) range-rate measurements leads to

$$\underbrace{\begin{bmatrix} -\dot{r}_1 \\ -\dot{r}_2 \\ \vdots \\ -\dot{r}_N \end{bmatrix}}_{\mathbf{y}} = \underbrace{\begin{bmatrix} \sin(\theta_1) & \cos(\theta_1) \\ \sin(\theta_2) & \cos(\theta_2) \\ \vdots & \\ \sin(\theta_N) & \cos(\theta_N) \end{bmatrix}}_{\mathbf{A}} \mathbf{h}_r^j, \tag{5.3}$$

and

$$\underbrace{\begin{bmatrix} -\dot{r}_1 \\ -\dot{r}_2 \\ \vdots \\ -\dot{r}_N \end{bmatrix}}_{\mathbf{y}} = \underbrace{\begin{bmatrix} \sin(\theta_1)\cos(\phi_1) & \cos(\theta_1)\cos(\phi_1) & \sin(\phi_1) \\ \sin(\theta_2)\cos(\phi_2) & \cos(\theta_2)\cos(\phi_2) & \sin(\phi_2) \\ \vdots & & \\ \sin(\theta_N)\cos(\phi_N) & \cos(\theta_N)\cos(\phi_N) & \sin(\phi_N) \end{bmatrix}}_{\mathbf{A}} \mathbf{h}_r^j, \tag{5.4}$$

for 2D and 3D radars, respectively. As shown in Stahoviak [2019], Doer and Trommer [2020b] and Kellner et al. [2015], one can recover the radar ego-velocity $\mathbf{h}_r$ using $N > d$ non-collinear, stationary landmarks, where d is 2 or 3 for 2D and 3D radars, respectively.

For both 2D and 3D range-rate models, the radar ego-velocity estimation problem is

$$\min_{\mathbf{h}_r^j} (\mathbf{y} - \mathbf{A}\mathbf{h}_r^j)^\intercal (\mathbf{y} - \mathbf{A}\mathbf{h}_r^j), \tag{5.5}$$

which is a standard least-squares optimization problem. As a result, the estimated ego-velocity at time j is

$$\mathbf{h}_r^{j\star} = (\mathbf{A}^\mathsf{T}\mathbf{A})^{-1}\mathbf{A}^\mathsf{T}\mathbf{y}, \tag{5.6}$$

with covariance

$$\mathbf{\Sigma}_r^j = \frac{(\boldsymbol{\epsilon}^\mathsf{T}\boldsymbol{\epsilon})(\mathbf{A}^\mathsf{T}\mathbf{A})}{N-d}, \tag{5.7}$$

where d is the dimension of $\mathbf{h}_r^j$.

Ego-velocity Estimation in Rigidly-Connected Reference Frames

Recall that in Section 2.5.3 we discussed the measurement of ego-velocity measurements relative to a rigidly-connected reference frame. Here, we attempt to estimate the 3D ego-velocity and rotational velocity of the rigidly-connected frame, $\underrightarrow{\mathcal{F}}_a$ using Equations (2.62) and (5.4). The 3D ego-velocity estimation model takes the form

$$\mathbf{y} = \underbrace{\mathbf{A}\mathbf{R}_{ar}{}^\mathsf{T}\begin{bmatrix}\mathbf{I}_{3\times 3} & -\mathbf{t}_a^{ra\wedge}\end{bmatrix}}_{\mathbf{A}_m}\begin{bmatrix}\mathbf{v}_a^{aw}(t)\\ \boldsymbol{\omega}_a^{aw}(t)\end{bmatrix}. \tag{5.8}$$

At first glance, the matrix $\mathbf{A}_m$ has N rows and six columns. Since N can be greater than six, we may consider using least squares to solve for the ego-velocity and rotational velocity of $\underrightarrow{\mathcal{F}}_a$. However, the subsequent problem is under-constrained because the modified matrix, $\mathbf{A}_m$, has a corank of 3. The null vectors of $\mathbf{A}_m$ are

$$\begin{aligned}&\frac{1}{\|\mathbf{t}_a^{ra}\|_2}\begin{bmatrix}\mathbf{0}\\ \mathbf{t}_a^{ra}\end{bmatrix},\ \frac{1}{\sqrt{\mathbf{x}^\mathsf{T}\mathbf{x}+\mathbf{y}^\mathsf{T}\mathbf{y}}}\begin{bmatrix}\mathbf{y}\\ \mathbf{x}\end{bmatrix},\ \frac{1}{\sqrt{(\mathbf{t}_a^{ra\mathsf{T}}\mathbf{t}_a^{ra})^2\mathbf{x}^\mathsf{T}\mathbf{x}+\mathbf{y}^\mathsf{T}\mathbf{y}}}\begin{bmatrix}(\mathbf{t}_a^{ra\mathsf{T}}\mathbf{t}_a^{ra})\mathbf{x}\\ -\mathbf{y}\end{bmatrix},\\ \text{s.t. }&\|\mathbf{x}\|_2 = 1,\\ &\mathbf{t}_a^{ra\mathsf{T}}\mathbf{x} = 0,\\ &\mathbf{y} = \mathbf{t}_a^{ra\wedge}\mathbf{x}.\end{aligned} \tag{5.9}$$

Intuitively, the null vectors are linear and rotational velocities of $\underrightarrow{\mathcal{F}}_a$ that do not induce any linear velocity in $\underrightarrow{\mathcal{F}}_r$. For example, if $\underrightarrow{\mathcal{F}}_a$ only rotates about the moment arm $\mathbf{t}_a^{ra}$, then the measured ego-velocity at $\underrightarrow{\mathcal{F}}_r$ is zero. Using this knowledge, we can solve for the velocities, but our estimate will be biased in the direction of the null vectors. Similar to the approach in Forster et al. [2017], we can pre-integrate the biased velocities, fuse with other sensor data, and estimate the biases as part of the problem. Or, we may increase the rank of $\mathbf{A}_m$ by fusing measurements from other radar sensors in the system, if available.

The equivalent 2D estimation model is trivial to derive from Equation (5.8). Using Equation (5.8) and the matrix $\mathbf{A}$ in Equation (5.3), the modified matrix for the 2D case, $\mathbf{A}_m$, is

$$\mathbf{A}_m = \mathbf{A}\mathbf{R}_{ar}{}^\mathsf{T}\begin{bmatrix}1 & 0 & -t_{a,y}^{ra}\\ 0 & 1 & t_{a,x}^{ra}\end{bmatrix}, \tag{5.10}$$

where the estimated velocity vector contains $v_{a,x}^{aw}$, $v_{a,y}^{aw}$, and $\omega_{a,z}^{aw}$. Similar to $\mathbf{A}_m$ in Equation (5.8), Equation (5.10)

contains the null vector

$$\frac{1}{\sqrt{1 + {t_{a,x}^{ra}}^2 + {t_{a,y}^{ra}}^2}} \begin{bmatrix} t_{a,y}^{ra} \\ -t_{a,x}^{ra} \\ 1 \end{bmatrix}. \tag{5.11}$$

To instantaneously estimate the ego-velocity and rotational velocity of the system, we must eliminate this null vector, which is one motivation for use of the multi-radar systems (see Chapter 6). Since Equations (5.8) and (5.10) involve the extrinsic parameters of the system, calibration processes that use these models must jointly estimate the calibration parameters, ego-velocity and rotational velocity of the system. When calibration problems include velocity estimation, the radar data rejection scheme we present in the following section cannot be used to reject radar data that does not 'fit' our models. Consequently, we prefer using the models defined by Equations (5.3) and (5.4).

Doppler Phenomenology and Outliers

Returning to the ego-velocity estimation problem defined by Equation (5.5), the radar data, $\mathcal{R}$, may contain outliers. The two main sources of outliers are targets that move relative to the inertial reference frame and spurious multipath reflections. Similar to Stahoviak [2019] and Doer and Trommer [2020b], we use random sample and concensus (RANSAC) and radar cross-section thresholding to remove outliers. Empirically, RANSAC eliminates outliers that do not 'fit' the models described by Equations (5.3) and (5.4). For some multipath reflections, outlier rejection using only RANSAC is insufficient; we elaborate on this below.

Generally, modelling the effects of multipath reflections on ego-velocity estimation requires simulating EM wave emission, propagation, reflection, and reception in a given environment. Fortunately, even a simplified radar and environmental model reveals that multipath reflections have the potential to introduce error into the range-rate measurement of a target and, subsequently, the ego-velocity estimate. To start, we assume that the radar and environment are entirely 2D. In the simplified radar model, we decompose the emitted radar wave into rays that travel radially away from the radar origin. The radar EM wave oscillates with a frequency of f and wavelength λ. At any point along the radar ray, we can compute the phase, ϕ, of the ray. Finally, an element of $\mathcal{R}$ is the result of the radar receiving a single return ray.

Let the radar antenna measure the return of a ray that travels along the path shown in Figure 5.3. The range and azimuth measurements for this path are $r = d_1 + d_2 + d_3$ and θ_2, respectively. If either landmark l_1 or l_2 are moving relative to the world frame, then the range-rate measurement is biased with respect to a stationary landmark at r and θ_2. If the radar receives true (non-multipath) returns from three stationary landmarks in

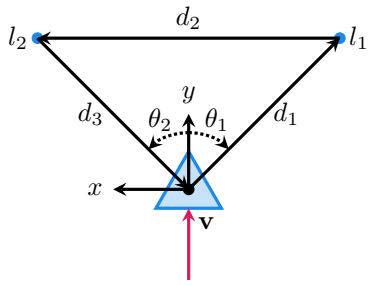


Figure 5.3: Diagram of a 2D multipath radar ray reflection. The path of the reflection is transceiver-l_1-l_2-receiver. The two blue blocks are the landmarks, where landmarks l_1 and l_2 are the left and right blocks, respectively. d_i are distances in meters, v is the velocity of the radar, and θ_i are angles in radians. The linear velocity of the system, $\mathbf{v}$, aligns with the y-axis in this diagram.

addition to the biased return, then the three returns form an inlier set and RANSAC is able to reject the outlier measurement. However, the effect of multipath error between two stationary landmarks on the range-rate measurement requires more investigation. Following the Doppler shift derivation in Rauber and Nesbitt [2018], we can define the time-varying phase of the electromagnetic wave at the transmitter as

$$\phi_t(t) = 2\pi f_t t, \tag{5.12}$$

where f_t is the emitted frequency of the radar wave. The phases of single and multipath reflections at the receiver are

$$\phi_{r,\text{single}}(t) = 2\pi \left(f_t t - 2\frac{d_3 - vt\cos(\theta_2)}{\lambda} \right), \tag{5.13}$$

$$\phi_{r,\text{multi}}(t) = 2\pi \left(f_t t - \frac{d_1 + d_2 + d_3 - vt\cos(\theta_1) - vt\cos(\theta_2)}{\lambda} \right), \tag{5.14}$$

where λ and v are the wavelength and speed of the radar, respectively. Rauber and Nesbitt [2018] model the electromagnetic wave frequency at the receiver as

$$f_r = \frac{1}{2\pi}\frac{\partial \phi_r(t)}{\partial t}, \tag{5.15}$$

As a result, the doppler shifts are

$$f_{r,\text{single}} = f_t + \frac{2v\cos(\theta_2)}{\lambda}, \tag{5.16}$$

$$f_{D,\text{single}} = \frac{2v\cos(\theta_2)}{\lambda}, \tag{5.17}$$

$$f_{r,\text{multi}} = f_t + \frac{v\left(\cos(\theta_1) + \cos(\theta_2)\right)}{\lambda}, \tag{5.18}$$

$$f_{D,\text{multi}} = \frac{v\left(\cos(\theta_1) + \cos(\theta_2)\right)}{\lambda}. \tag{5.19}$$

Figure 5.4 shows the percentage doppler shift error for the multipath reflection relative to a single reflection. We expect the symmetry in Figure 5.4 because the Doppler shift is a function of the radial velocity. For a fixed landmark azimuth, we can generate a radial velocity with two possible radar ego-velocity vectors (see Chapter 8.9 in Richards et al. [2010]). That is, if the radar ray originates at angle θ_1 and $\cos(\theta_1) = \cos(\theta_2)$, then the range-rate value fits the model even though there is no landmark at $\begin{bmatrix} r\sin(\theta_2) & r\cos(\theta_2) \end{bmatrix}$. Further, the percentage doppler shift error is minimal for multipath reflections with small differences between the transmission and return angles, or angles that are close to the velocity of the radar. After empirically tuning the RANSAC parameters, we found that RANSAC rejects multipath reflections that deviate from Equations (5.3) and (5.4) by greater than 2.5%. While some multipath reflections might still be labelled as inliers by RANSAC, these reflections have a lower radar cross-section relative to single reflections from similar locations and can be rejected by radar cross-section thresholding. Finally, our discussion here greatly simplifies the (true) interactions between the radar EM wave and the environment. A more in-depth simulation could be performed to glean a deeper understanding of how multipath reflections affect ego-velocity estimation.

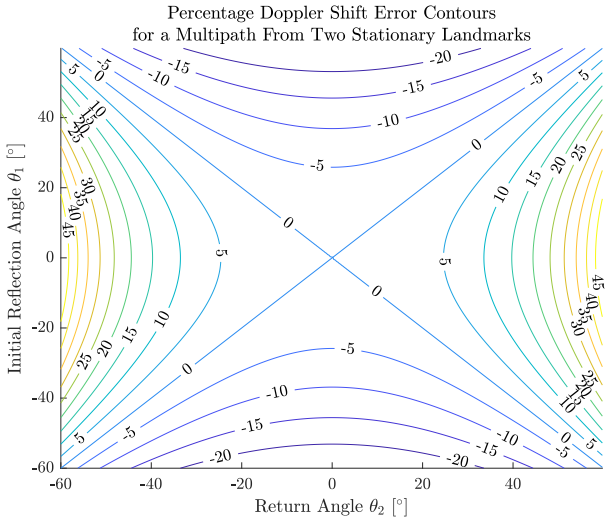


Figure 5.4: Percentage doppler shift error for a multipath reflection relative to a single reflection. Multipath reflections that result in a large change in the reflection and return angles induce a large error in the doppler shift, relative to the single path reflections. In radar ego-velocity estimation, these reflections are rejected by RANSAC. If the difference between the initial reflection and return angles of a multipath reflection is small, then the doppler shift error is minimal. Additionally, the doppler shift error is minimal if the initial reflection and return angles are symmetric about the direction of motion of the radar.

5.3 Problem Formulation

Let $\underrightarrow{\mathcal{F}}_r$, $\underrightarrow{\mathcal{F}}_c$, and $\underrightarrow{\mathcal{F}}_w$ represent the 3D mm-wavelength radar, monocular camera, and stationary world coordinate frames, respectively. As shown in Chiuso et al. [2002], we can use observed, stationary landmarks to estimate the scaled pose of the camera with respect to $\underrightarrow{\mathcal{F}}_w$, up to an unknown translational scale factor α.

$$\begin{aligned}
\mathbf{R}_{cw}(t_s) &= \exp(\mathbf{n}_r)\mathbf{R}_{cr}\mathbf{R}_{wr}(t_s)^\mathsf{T}, \\
\mathbf{n}_r &\sim \mathcal{N}\left(\mathbf{0}_{3\times1}, \mathbf{\Sigma}_r\right), \\
\mathbf{t}_c^{wc}(t_s) &= \alpha(\mathbf{R}_{cr}\mathbf{t}_r^{wr}(t_s) + \mathbf{t}_c^{rc}) + \mathbf{n}_t, \\
\mathbf{n}_t &\sim \mathcal{N}\left(\mathbf{0}_{3\times1}, \mathbf{\Sigma}_t\right),
\end{aligned} \tag{5.20}$$

where $\mathbf{n}_r$ and $\mathbf{n}_t$ are zero-mean Gaussian noise terms for the camera rotation and translation measurements, respectively, with covariances matrices $\mathbf{\Sigma}_r$ and $\mathbf{\Sigma}_t$. Additionally, the time-varying translation, $\mathbf{t}_r^{wr}(t_s)$, and rotation, $\mathbf{R}_{wr}(t_s)$, are $\mathbb{R}^3$ and $\mathrm{SO}(3)$ B-splines of order four. Consequently, the camera pose error distributions are

$$\mathbf{e}_{r,s} = \log(\mathbf{R}_{cw}(t_s)\mathbf{R}_{wr}(t_s)\mathbf{R}_{cr}{}^\mathsf{T})^\vee \sim \mathcal{N}\left(\mathbf{0}_{3\times1}, \mathbf{\Sigma}_r\right), \tag{5.21}$$

$$\mathbf{e}_{t,s} = \mathbf{t}_c^{wc}(t_s) - \alpha(\mathbf{R}_{cr}\mathbf{t}_r^{wr}(t_s) + \mathbf{t}_c^{rc}) \sim \mathcal{N}\left(\mathbf{0}_{3\times1}, \mathbf{\Sigma}_t\right). \tag{5.22}$$

Given the measurement time, t_s, is $t_i \leq t_s < t_{i+1}$ and the normalized time is u_s, the nonzero partial derivatives of the error equations are

$$\frac{\partial \mathbf{e}_{r,s}}{\partial \boldsymbol{\psi}_{cr}} \approx -\mathbf{I}, \tag{5.23}$$

$$\frac{\partial \mathbf{e}_{r,s}}{\partial \boldsymbol{\psi}_i} \approx \mathbf{R}_{cr}\mathbf{R}_{wr}(t_s)^\intercal, \tag{5.24}$$

$$\frac{\partial \mathbf{e}_{r,s}}{\partial \boldsymbol{\phi}_j^i} \approx \boldsymbol{\lambda}_j(u_s)\mathbf{R}_{cr}\left(\prod_{l=j+1}^{3}\exp((\boldsymbol{\lambda}_l(u_s)\boldsymbol{\phi}_l^i)^\wedge)\right)^\intercal \mathbf{J}_r(\boldsymbol{\lambda}_j(u_s)\boldsymbol{\phi}_j^i), \tag{5.25}$$

$$\frac{\partial \mathbf{e}_{t,s}}{\partial \alpha} = -(\mathbf{R}_{cr}\mathbf{t}_r^{wr}(t_s) + \mathbf{t}_c^{rc}), \tag{5.26}$$

$$\frac{\partial \mathbf{e}_{t,s}}{\partial \boldsymbol{\psi}_{cr}} \approx \alpha(\mathbf{R}_{cr}\mathbf{t}_r^{wr}(t_s))^\wedge, \tag{5.27}$$

$$\frac{\partial \mathbf{e}_{t,s}}{\partial \mathbf{t}_i} = -\alpha(1 - \boldsymbol{\lambda}_1(u_s))\mathbf{R}_{cr}, \tag{5.28}$$

$$\frac{\partial \mathbf{e}_{t,s}}{\partial \mathbf{t}_{i+j}} = -\alpha(\boldsymbol{\lambda}_j(u_s) - \boldsymbol{\lambda}_{j+1}(u_s))\mathbf{R}_{cr},\ \forall\, j = 1, \ldots, 3 \tag{5.29}$$

$$\frac{\partial \mathbf{e}_{t,s}}{\partial \mathbf{t}_c^{rc}} = -\alpha\mathbf{I}, \tag{5.30}$$

where we assume the errors $\mathbf{e}_{r,s}$ and $\mathbf{e}_{t,s}$ have a small magnitude. From our review of B-splines in Chapter 2, we recall that $\boldsymbol{\lambda}(u) = \tilde{\mathbf{M}}_k\mathbf{u}$, $\tilde{\mathbf{M}}_k$ is a mixing matrix, $\mathbf{u}^T = [1 \;\; u \;\; u^2 \;\; \ldots \;\; u^{k-1}]$, and k is the order of the B-spline. We note that a monocular visual odometry (VO) algorithm (i.e., localization without loop closure) could provide camera ego-motion measurements, but visual drift will bias these measurements and decrease calibration accuracy. That said, the radar ego-velocity is a local property of a trajectory and our approach does not mitigate pose errors induced by visual drift.

We use an ego-velocity sensor as an abstracted representation of the radar unit. For spatiotemporal calibration, we model the radar ego-velocity as

$$\begin{aligned}
\mathbf{v}_r(t_s) &= -\dot{\mathbf{t}}_r^{wr}(t_s+\tau) - \boldsymbol{\omega}_r^{rw}(t_s+\tau)^\wedge\mathbf{t}_r^{wr}(t_s+\tau) + \mathbf{n}_v(t_s+\tau),\\
\mathbf{n}_v &\sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_v(t_s+\tau)),\\
\mathbf{e}_{v,s} &= \mathbf{v}_r(t_s) + \dot{\mathbf{t}}_r^{wr}(t_s+\tau) + \boldsymbol{\omega}_r^{rw}(t_s+\tau)^\wedge\mathbf{t}_r^{wr}(t_s+\tau) \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_v(t_s+\tau)).
\end{aligned} \tag{5.31}$$

In Equation (5.31), τ is the temporal offset of the radar measurements relative to the camera measurements and $\mathbf{n}_v$ is the radar ego-velocity measurement noise term that we assume is a zero-mean Gaussian with covariance matrix $\boldsymbol{\Sigma}_v$ (see Equation (5.7)). We choose to model the radar measurements as temporally offset from the camera measurements because the derivative of Equation (5.21) with respect to a time offset is clearly more complex than computing the derivative of Equation (5.31) with respect to τ. Importantly, modelling the camera measurements as temporally offset from the radar measurements is equally valid. Additionally, we choose to use the robocentric ego-velocity model, but all models in Section 2.5.3 are equally valid. Given the measurement time, $t_s + \tau$, is $t_i \leq t_s + \tau < t_{i+1}$ and the normalized time is u_s, the nonzero Jacobians are

$$\frac{\partial \mathbf{e}_{v,s}}{\partial \mathbf{t}_i} = -\frac{\dot{\boldsymbol{\lambda}}_1(u_s)}{\Delta t}\mathbf{I} + (1 - \boldsymbol{\lambda}_1(u_s))\boldsymbol{\omega}_r^{rw}(t_s+\tau)^\wedge, \tag{5.32}$$

$$\frac{\partial \mathbf{e}_{v,s}}{\partial \mathbf{t}_{i+j}} = \frac{(\dot{\boldsymbol{\lambda}}_j(u_s) - \dot{\boldsymbol{\lambda}}_{j+1}(u_s))}{\Delta t}\mathbf{I} + (\boldsymbol{\lambda}_j(u_s) - \boldsymbol{\lambda}_{j+1}(u_s))\boldsymbol{\omega}_r^{rw}(t_s+\tau)^\wedge, \tag{5.33}$$

$$\frac{\partial \mathbf{e}_{v,s}}{\partial \boldsymbol{\phi}_i^j} = -\mathbf{t}_r^{wr}(t_s+\tau)^\wedge\frac{\partial \boldsymbol{\omega}_r^{rw}(t_s+\tau)}{\partial \boldsymbol{\phi}_i^j}, \tag{5.34}$$

$$\frac{\partial \mathbf{e}_{v,s}}{\partial \tau} = \ddot{\mathbf{t}}_r^{wr}(t_s+\tau) + \boldsymbol{\omega}_r^{rw}(t_s+\tau)^\wedge\dot{\mathbf{t}}_r^{wr}(t_s+\tau) - \mathbf{t}_r^{wr}(t_s+\tau)^\wedge\dot{\boldsymbol{\omega}}_r^{rw}(t_s+\tau). \tag{5.35}$$

Using a recurrence relation (from $j = k-1, \ldots, 1$), Sommer et al. [2020] defines $\frac{\partial \boldsymbol{\omega}_r^{rw}(t)}{\partial \boldsymbol{\phi}_i^j}$ as

$$\frac{\partial \boldsymbol{\omega}_r^{rw}(t)}{\partial \boldsymbol{\phi}_i^j} = \mathbf{P}_j \frac{\partial \boldsymbol{\omega}_{r,(j+1)}^{rw}(t)}{\partial \boldsymbol{\phi}_i^j}, \tag{5.36}$$

$$\mathbf{P}_{k-1} = \mathbf{I}, \tag{5.37}$$

$$\mathbf{P}_{j-1} = \mathbf{P}_j \exp(-\lambda_j(u)\boldsymbol{\phi}_i^j), \tag{5.38}$$

$$\frac{\partial \boldsymbol{\omega}_{r,(j+1)}^{rw}(t)}{\partial \boldsymbol{\phi}_i^j} = \lambda_j(u) \exp(-\lambda_j(u)\boldsymbol{\phi}_i^j)(\boldsymbol{\omega}_{r,(j)}^{rw}(t))^\wedge \mathbf{J}_r(-\lambda_j(u)\boldsymbol{\phi}_i^j) + \dot{\lambda_j}\mathbf{I}, \tag{5.39}$$

where u is the normalized time.

The set of parameters $\mathbf{x}$ that we wish to estimate are the spline control points ($\mathbf{t}_{0\ldots N} \in \mathbb{R}^3$, $\mathbf{R}_{0\ldots N} \in \mathrm{SO}(3)$), the extrinsic calibration parameters ($\mathbf{R}_{cr}, \mathbf{t}_c^{rc}$), the camera translation scale factor (α), and the temporal offset (τ),

$$\mathbf{x} = \begin{Bmatrix} \mathbf{t}_0, & \ldots, & \mathbf{t}_N, & \mathbf{R}_0, & \ldots, & \mathbf{R}_N, & \mathbf{R}_{cr}, & \mathbf{t}_c^{rc}, & \alpha, & \tau \end{Bmatrix}. \tag{5.40}$$

Given N_r radar measurements and N_c camera measurements, our optimization problem is as follows.

Problem 10. *Radar-Camera Spatiotemporal Calibration Problem*

$$\min_{\mathbf{x}} \sum_{s=1}^{N_r} \mathbf{e}_{v,s}{}^\intercal {\boldsymbol{\Sigma}_{v,s}}^{-1} \mathbf{e}_{v,s} + \sum_{s=1}^{N_c} \mathbf{e}_{r,s}{}^\intercal \boldsymbol{\Sigma}_r^{-1} \mathbf{e}_{r,s} + \mathbf{e}_{t,s}{}^\intercal \boldsymbol{\Sigma}_t^{-1} \mathbf{e}_{t,s}. \tag{5.41}$$

We perform this minimization using the Ceres solver from Agarwal et al. [2022], which is a standard non-linear least squares solver. The ability to calibrate all of the relevant parameters depends upon the identifiability of problem. In the next subsection, we discuss in the nonlinear observability of the 3D mm-wavelength radar-camera extrinsic calibration problem, which is then followed by a subsection on the identifiability of 3D mm-wavelength radar-camera spatiotemporal calibration.

5.4 Observability of Extrinsic Calibration

In this subsection, we demonstrate that radar-to-camera spatial calibration is locally weakly observable. Our observability proof employs the unit quaternion rotation representation rather than the Lie group representation of rotations (see Section 5.3). Since the unit quaternion constraint and map from unit quaternions to rotation matrices are both polynomials, the unit quaternion representation simplifies our mathematical analysis. In this subsection, we make use of the procedure from Hermann and Krener [1977]. Finally, we discuss one possible degenerate motion of the system, for which calibration is not possible.

5.4.1 Local Weak Observability

Following the procedure outlined in Hermann and Krener [1977], we define the system equations, compute the respective Lie derivatives, and show that the nonlinear observability matrix has full column rank. In our analysis here, the pose, velocity, and acceleration states of the radar-camera system are camera-centric (i.e., taken with respect to the camera and not the radar). Since the camera-centric states can be used to determine the radar-centric states, this change does not affect the observability result.

Given the camera frame $\underrightarrow{\mathcal{F}}_c$, the world frame $\underrightarrow{\mathcal{F}}_w$, and the radar frame $\underrightarrow{\mathcal{F}}_r$, the state vector for the observability analysis is defined as

$$\mathbf{x} = \begin{bmatrix} \mathbf{t}_w^{cw\intercal} & \mathbf{q}_{wc}^{\intercal} & \mathbf{v}_w^{cw\intercal} & \boldsymbol{\omega}_c^{cw\intercal} & \mathbf{a}_w^{cw\intercal} & \boldsymbol{\alpha}_c^{cw\intercal} & \alpha & \mathbf{t}_c^{rc\intercal} & \mathbf{q}_{cr}^{\intercal} \end{bmatrix}^{\intercal}, \tag{5.42}$$

where $\mathbf{t}$, $\mathbf{v}$, and $\mathbf{a}$ denote the translation, linear velocity, and linear acceleration. The vectors $\boldsymbol{\omega}$ and $\boldsymbol{\alpha}$ are the rotational velocity and the rotational acceleration, respectively. Finally, α is the scale factor for the camera translation (for a monocular camera system). The motion model for the system is

$$\dot{\mathbf{x}} = \mathbf{f}_0(\mathbf{x}) + \mathbf{f}_1(\mathbf{x}) = \begin{bmatrix} \mathbf{0}_{3\times1} \\ \frac{1}{2}\Xi(\mathbf{q}_{wc})\,\boldsymbol{\omega}_c^{cw} \\ \mathbf{0}_{3\times1} \\ \boldsymbol{\alpha}_c^{cw} \\ \mathbf{0}_{3\times1} \\ \mathbf{0}_{3\times1} \\ 0 \\ \mathbf{0}_{3\times1} \\ \mathbf{0}_{4\times1} \end{bmatrix} + \begin{bmatrix} \mathbf{v}_w^{cw} \\ \mathbf{0}_{4\times1} \\ \mathbf{a}_w^{cw} \\ \mathbf{0}_{3\times1} \\ \mathbf{0}_{3\times1} \\ \mathbf{0}_{3\times1} \\ 0 \\ \mathbf{0}_{3\times1} \\ \mathbf{0}_{4\times1} \end{bmatrix}. \tag{5.43}$$

The measurement model equations for the (scaled) camera translation and rotation are, respectively,

$$\mathbf{h}_1 = \alpha\,\mathbf{t}_w^{cw}, \tag{5.44}$$

$$\mathbf{h}_2 = \mathbf{q}_{wc}. \tag{5.45}$$

Finally, the radar ego-velocity measurement equation is

$$\mathbf{h}_3 = \mathbf{R}^{\intercal}(\mathbf{q}_{cr})(\mathbf{R}^{\intercal}(\mathbf{q}_{wc})\mathbf{v}_w^{cw} + \boldsymbol{\omega}_c^{cw\wedge}\mathbf{t}_c^{rc}). \tag{5.46}$$

The observability analysis requires the zeroth-, first-, and second-order Lie derivatives. The zeroth-order Lie derivatives are

$$\begin{aligned} \nabla L^0\mathbf{h}_1 &= \begin{bmatrix} \alpha\mathbf{I}_3 & \mathbf{0}_{3\times16} & \mathbf{t}_w^{cw} & \mathbf{0}_{3\times7} \end{bmatrix}, \\ \nabla L^0\mathbf{h}_2 &= \begin{bmatrix} \mathbf{0}_{4\times3} & \mathbf{I}_4 & \mathbf{0}_{4\times20} \end{bmatrix}, \\ \nabla L^0\mathbf{h}_3 &= \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{A} & \mathbf{R}^{\intercal}(\mathbf{q}_{cr})\mathbf{R}^{\intercal}(\mathbf{q}_{wc}) & -\mathbf{R}^{\intercal}(\mathbf{q}_{cr})\mathbf{t}_c^{rc\wedge} & \mathbf{0}_{3\times7} & \mathbf{R}^{\intercal}(\mathbf{q}_{cr})\boldsymbol{\omega}_c^{cw\wedge} & \mathbf{B} \end{bmatrix}, \end{aligned} \tag{5.47}$$

where

$$\begin{aligned} \mathbf{A} &= \mathbf{R}^{\intercal}(\mathbf{q}_{cr})\frac{\partial\mathbf{R}^{\intercal}(\mathbf{q}_{wc})\mathbf{v}_w^{cw}}{\partial\mathbf{q}_{wc}}, \\ \mathbf{B} &= \frac{\partial\mathbf{R}^{\intercal}(\mathbf{q}_{cr})(\mathbf{R}^{\intercal}(\mathbf{q}_{wc})\mathbf{v}_w^{cw} + \boldsymbol{\omega}_c^{cw\wedge}\mathbf{t}_c^{rc})}{\partial\mathbf{q}_{cr}}. \end{aligned} \tag{5.48}$$

The first-order Lie derivatives are

$$\begin{aligned}
\nabla L^1_{\mathbf{f}_1}\mathbf{h}_1 &= \begin{bmatrix}\mathbf{0}_{3\times7} & \alpha\mathbf{I}_3 & \mathbf{0}_{3\times9} & \mathbf{v}_w^{cw} & \mathbf{0}_{3\times7}\end{bmatrix},\\
\nabla L^1_{\mathbf{f}_0}\mathbf{h}_2 &= \begin{bmatrix}\mathbf{0}_{4\times3} & \frac{1}{2}\Omega(\boldsymbol{\omega}_c^{cw}) & \mathbf{0}_{4\times3} & \frac{1}{2}\Xi(\mathbf{q}_{wc}) & \mathbf{0}_{4\times14}\end{bmatrix},\\
\nabla L^1_{\mathbf{f}_0}\mathbf{h}_3 &= \begin{bmatrix}\mathbf{0}_{3\times3} & \mathbf{C} & \mathbf{D} & \mathbf{E} & \mathbf{0}_{3\times3} & \mathbf{F} & \mathbf{0}_{3\times1} & \mathbf{R}^\intercal(\mathbf{q}_{cr})\boldsymbol{\alpha}_c^{cw\wedge} & \mathbf{G}\end{bmatrix},\\
\nabla L^1_{\mathbf{f}_1}\mathbf{h}_3 &= \begin{bmatrix}\mathbf{0}_{3\times3} & \mathbf{H} & \mathbf{0}_{3\times6} & \mathbf{R}^\intercal(\mathbf{q}_{cr})\mathbf{R}^\intercal(\mathbf{q}_{wc}) & \mathbf{0}_{3\times7} & \mathbf{L}\end{bmatrix},
\end{aligned} \tag{5.49}$$

where

$$\begin{aligned}
\mathbf{H} &= \mathbf{R}^\intercal(\mathbf{q}_{cr})\frac{\partial\mathbf{R}^\intercal(\mathbf{q}_{wc})\mathbf{a}_w^{cw}}{\partial\mathbf{q}_{wc}},\\
\mathbf{L} &= \frac{\partial\mathbf{R}^\intercal(\mathbf{q}_{cr})\mathbf{R}^\intercal(\mathbf{q}_{wc})\mathbf{a}_w^{cw}}{\partial\mathbf{q}_{cr}}.
\end{aligned} \tag{5.50}$$

We do not explicitly require the matrices $\mathbf{C}$, $\mathbf{E}$, and $\mathbf{F}$ in Equation (5.49) because the submatrix formed from the columns corresponding to the rotation states can be shown to be full rank. The matrices $\mathbf{D}$ and $\mathbf{G}$ are required for the analysis, but we omit them here for brevity. The second order Lie derivatives are

$$\begin{aligned}
\nabla L^2_{\mathbf{f}_1}\mathbf{h}_1 &= \begin{bmatrix}\mathbf{0}_{3\times13} & \alpha\mathbf{I}_3 & \mathbf{0}_{3\times3} & \mathbf{a}_w^{cw} & \mathbf{0}_{3\times7}\end{bmatrix},\\
\nabla L^2_{\mathbf{f}_0}\mathbf{h}_2 &= \begin{bmatrix}\mathbf{0}_{4\times3} & \frac{1}{4}(2\boldsymbol{\Omega}(\boldsymbol{\alpha}_c^{cw}) - \boldsymbol{\omega}_c^{cw\intercal}\boldsymbol{\omega}_c^{cw}\mathbf{I}_4) & \mathbf{0}_{4\times3} & -\frac{1}{2}\mathbf{q}_{wc}\boldsymbol{\omega}_c^{cw\intercal} & \mathbf{0}_{4\times3} & \frac{1}{2}\Xi(\mathbf{q}_{wc}) & \mathbf{0}_{4\times8}\end{bmatrix}.
\end{aligned} \tag{5.51}$$

Stacking the gradients of the Lie derivatives, we arrive at the nonlinear observability matrix,

$$\mathbf{O} = \begin{bmatrix}
\nabla L^0\mathbf{h}_1\\
\nabla L^1_{f_1}\mathbf{h}_1\\
\nabla L^2_{f_1f_1}\mathbf{h}_1\\
\nabla L^0\mathbf{h}_2\\
\nabla L^1_{f_0}\mathbf{h}_2\\
\nabla L^2_{f_0f_0}\mathbf{h}_2\\
\nabla L^0\mathbf{h}_3\\
\nabla L^1_{f_0}\mathbf{h}_3\\
\nabla L^1_{f_1}\mathbf{h}_3
\end{bmatrix}. \tag{5.52}$$

This matrix can be shown to have full column rank, and hence the system is locally weakly observable.[1].

5.4.2 Degenerate Motions

Loss of observability will occur when $\mathbf{O}$ in Equation (5.56) does not have full column rank. This proof relies on non-zero linear and rotational velocities and linear accelerations, so the matrix $\mathbf{O}$ clearly loses full rank when the radar-camera platform is not translating, rotating, or accelerating. Additionally, the system must rotate about more than one axis. We expect this degeneracy because our problem is similar to the one defined by Brookshire and Teller [2012]. We posit, based on our experiments, that degenerate motions occur infrequently in practice.

[1] Additional details are in Appendix A

5.5 Identifiability of 3D Radar-Camera Spatiotemporal Calibration

In this section, we show that the calibration problem is identifiable given sufficient excitation of the radar-camera system. Our approach is to determine the observability, or 'instantaneous identifiability,' of the system at several different points in time, assuming that the system follows a varying trajectory. Using the rank condition in Section 4.3.2, we consider local identifiability (cf. locally weak observability) along a trajectory segment in this section. A similar approach has been taken in Li and Mourikis [2014] and Hewitt and Marshall [2015] and elsewhere. Following the local identifiability proof, we prove that the calibration problem is weakly identifiable using the procedure in Section 4.3.2. Finally, we describe several 'degenerate' motions for which the identifiability condition does not hold.

5.5.1 Identifiability of Radar-Camera Spatiotemporal Calibration

We begin by simplifying the state (and parameter) vector that we aim to estimate. Since we are working in continuous time (or, roughly equivalently, if there are a sufficient number of closely-spaced radar and camera measurements), the scaled ego-velocity of the camera in the camera reference frame $\alpha\mathbf{v}_c^{cw}(t_i)$, the rotational velocity of the camera relative to the inertial frame in the camera frame $\boldsymbol{\omega}_c(t_i)$, the radar ego-velocity $\mathbf{v}_r(t_i+\tau)$, and the time derivative of the radar ego-velocity $\dot{\mathbf{v}}_r(t_i+\tau)$ are all available. For the purposes of identifiability, we are able to define the following, modified measurement model,

$$\mathbf{h}(t_i) = \alpha(\mathbf{R}_{cr}\mathbf{v}_r(t_i+\tau) - \boldsymbol{\omega}_c(t_i)^\wedge \mathbf{t}_c^{rc}), \tag{5.53}$$

where $\mathbf{h}(t_i)$ is the scaled linear velocity of the camera ($\mathbf{v}_c^{cw}$) and $\boldsymbol{\omega}_c$ is the rotational velocity of the camera, both relative to the camera frame. This modified measurement model does not directly rely on the pose of the radar, thus simplifying the set of parameters that we wish to determine to

$$\tilde{\mathbf{x}} = \{\mathbf{t}_c^{rc}, \quad \mathbf{R}_{cr}, \quad \alpha, \quad \tau\}. \tag{5.54}$$

To decrease the notational burden, we drop the superscripts and subscripts defining the velocities and extrinsic transform parameters. The gradient of the zeroth-order Lie derivative of the ith measurement is

$$\nabla_{\tilde{\mathbf{x}}} L_0\mathbf{h}(t_i) = \begin{bmatrix} -\alpha\boldsymbol{\omega}(t_i)^\wedge & -\alpha(\mathbf{R}\mathbf{v}(t_i+\tau))^\wedge\mathbf{J}_l & \mathbf{R}\mathbf{v}(t_i+\tau) - \boldsymbol{\omega}(t_i)^\wedge\mathbf{t} & \alpha\mathbf{R}\dot{\mathbf{v}}(t_i+\tau) \end{bmatrix}, \tag{5.55}$$

where $\mathbf{J}_l$ is the Lie algebra left Jacobian of $\mathbf{R}_{cr}$. Since the parameters of interest are constant with respect to time, we are able to stack the gradients of several Lie derivatives (at different points in time) to form the observability matrix,

$$\mathbf{O} = \begin{bmatrix} \nabla_{\tilde{\mathbf{x}}} L_0\mathbf{h}(t_1) \\ \nabla_{\tilde{\mathbf{x}}} L_0\mathbf{h}(t_2) \\ \nabla_{\tilde{\mathbf{x}}} L_0\mathbf{h}(t_3) \end{bmatrix}, \tag{5.56}$$

which, using block Gaussian elimination, can be shown to have full column rank when three or more sets of measurements are available.

Two comments regarding the analysis are in order. First, we note that the analysis is simplified by considering the modified measurement equation only (without any higher-order Lie derivatives). Second, there is a subtlety involved in stacking the gradients of the Lie derivatives at different points in time. The modified measurement equation depends upon the time derivatives of the camera pose and the radar ego-velocity—this implies that,

although we do not consider specific control inputs, the system dynamics must be non-null. Stated differently, varied motion of the radar-camera pair is necessary to ensure identifiability; we discuss this requirement further in this section. Also, it is worth noting that, as shown in Anguelova [2007] and Kelly et al. [2021], the observation times must span the temporal offset period.

5.5.2 Degenerate Motions

The conditions under which a loss of observability (identifiability) may occur can be determined by examining the nullspace of Equation (5.56). Here, we consider the scale parameter and temporal parameter to be known, which removes the last column of the matrix defined by Equation (5.55)—in turn, only two sets of measurements are required. The nullspace of $\nabla_{\mathbf{x}} L_0 \mathbf{h}(t_i)$ contains the vectors

$$\mathbf{U}_i = \begin{bmatrix} \boldsymbol{\omega}(t_i) & \mathbf{0} & (\mathbf{I} - \frac{\boldsymbol{\omega}(t_i)\boldsymbol{\omega}(t_i)^\intercal}{\|\boldsymbol{\omega}(t_i)\|^2})\mathbf{R}\mathbf{v}(t_i) \\ \mathbf{0} & \mathbf{J}_l^{-1}\mathbf{R}\mathbf{v}(t_i) & (\mathbf{I} - \frac{\mathbf{J}_l^{-1}\mathbf{R}\mathbf{v}(t_i)(\mathbf{J}_l^{-1}\mathbf{R}\mathbf{v}(t_i))^T}{\left\|\mathbf{J}_l^{-1}\mathbf{R}\mathbf{v}(t_i)\right\|^2})\mathbf{J}_l^{-1}\boldsymbol{\omega}(t_i) \end{bmatrix}, \tag{5.57}$$

where each column of $\mathbf{U}_i$ defines one null vector. To ensure that the stacked observability matrix formed from $\nabla_{\mathbf{x}} L_0 \mathbf{h}(t_1)$ and $\nabla_{\mathbf{x}} L_0 \mathbf{h}(t_2)$ has full column rank (i.e., that the nullspace contains the zero vector only), the following constraints must be satisfied, at minimum:

$$\begin{aligned} \boldsymbol{\omega}(t_2) \times \boldsymbol{\omega}(t_1) &\neq 0, \\ \mathbf{v}(t_2) \times \mathbf{v}(t_1) &\neq 0. \end{aligned} \tag{5.58}$$

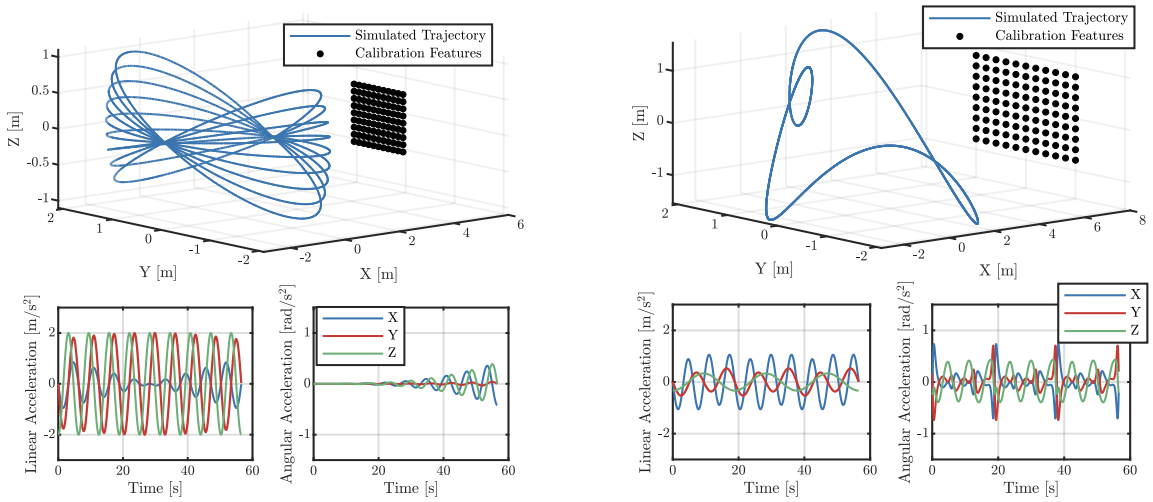
The constraints defined by Equation (5.58) show that the system must rotate about and translate along two non-collinear axes at different points in time. The rotation constraint is expected from our prior analysis. However, the rotational velocity of the radar unit cannot be measured directly, which leads to the second excitation requirement. This result is particularly interesting as recovering the second constraint using the criterion from Hermann and Krener [1977] is difficult. We note that the radar ego-acceleration, $\dot{\mathbf{v}}_r(t_i + \tau)$, *can* be constant for all i without reducing the column rank of Equation (5.55). Additional constraints can be generated from the third column of Equation (5.57), but these motions are more difficult to characterize; we posit, based on our experiments, that these constraints are less likely to be violated in practice. Finally, the addition of the temporal offset to the system has the potential to introduce additional degenerate motions, although we have not found any thus far that would impact real-world calibration.

5.6 Experiments

Sections 5.6.1 and 5.6.2 present two different simulation studies and four real-world experiments, respectively. The four real-world experiments are performed in a variety of environments.

5.6.1 Simulation Studies

We carried out a series of simulation studies to test the robustness of our algorithm to measurement noise. We generated a pair simulated camera-radar datasets using two different trajectories and varying noise levels (see Figures 5.5a and 5.5b). The nominal (noise-free) trajectories were selected to ensure sufficient excitation of



(a) High linear and low rotational velocity trajectory.

(b) Low linear and high rotational velocity trajectory.

Figure 5.5: Trajectories for the simulated experiments with associated linear and rotational acceleration plots.

the camera-radar pair for calibration. The median linear and rotational velocities for the trajectory shown in Figure 5.5a were, respectively, higher and lower than the velocities of the trajectory shown in Figure 5.5b.

We computed the simulated radar ego-velocity and camera pose measurements at points along the nominal trajectories. Since radar measurements are dependent on antenna-configuration and environment-specific, we did not simulate radar data at the electromagnetic propagation level. Simulated radar ego-velocity measurements (i.e., $\mathbf{v}_r(t_s)$) were computed using the known linear and rotational velocities defined by the trajectory. Consequently, our simulated radar measurements generalize to any radar and environment that produce an unbiased 3D ego-velocity estimate. Simulated camera pose measurements (i.e., $\mathbf{R}_{cw}(t_k)$ and $\mathbf{t}_c^{wc}(t_k)$) were derived from observations of a series of landmark points, arranged in a 2D grid. This configuration of points matches the configuration of a standard 'checkerboard' camera calibration target. In the targetless setting, we can only estimate the position of the camera up to an unknown scale, as shown in Chiuso et al. [2002]. To simulate a pose of unknown scale, the checkerboard tracking algorithm is given an incorrect size for the checkerboard squares and our calibration algorithm must estimate the α that scales the checkerboard squares to the correct size.

For each simulation, we added zero-mean Gaussian noise to the radar ego-velocity measurements ($\mathbf{\Sigma}_{v_s} = \sigma_r^2\,\mathbf{I}_{3\times3}$) and to the camera measurements of the checkerboard corners on the simulated image plane ($\mathbf{\Sigma}_{p_k} = \sigma_c^2\,\mathbf{I}_{2\times2}$). For our experiments, we adjusted the radar ego-velocity variance (σ_r) between 0.05 m/s and 0.15 m/s. Based on our real-world experiments (see Section 5.6.2), we have found that the radar ego-velocity measurement noise is closer to the lower end of this range, unless the environment is sparse and too few valid radar returns are received. We adjusted the variance the of noise added to the measured checkerboard corner coordinates (σ_c) between 0.1 and 0.4 pixels; these noise levels are similar to the observed noise in our real-world experiments Wise et al. [2021].

The error distributions for the spatial calibration parameter estimates ($\mathbf{R}_{cr}, \mathbf{t}_c^{rc}$), scale factor ($\alpha$), and temporal offset ($\tau$) are shown in Figures 5.6 and 5.7, across the 100 simulation trials. For the high linear and low rotational velocity trajectory, even in the high-noise regime, the error in the rotation and scale estimates remains at less than two degrees and one percent, respectively. However, high levels of noise in the radar ego-velocity measurements result in substantially larger (and more widely distributed) errors in the estimate of the relative translation of the sensors and of the temporal offset; the errors can be as large as 15 cm and 30 ms, respec-

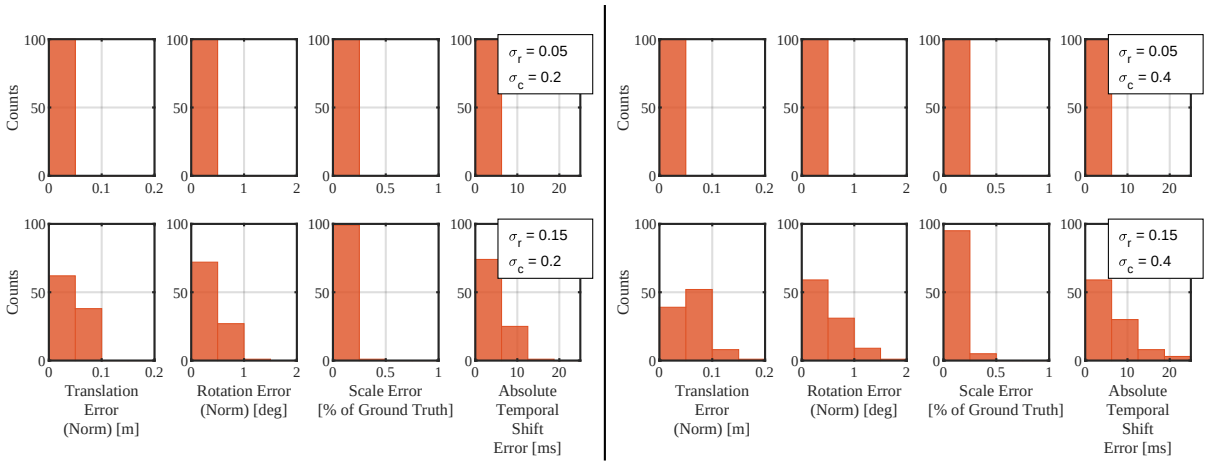


Figure 5.6: High linear and low rotational velocity trajectory spatiotemporal calibration results from the simulated experiments. Each subplot is a histogram of the error between the estimated and true parameter values for 100 experiments at a given level of measurement noise. Each row presents the results for a level of measurement noise. The levels of measurement noise are a combination of two radar measurement noise levels ($\sigma_r = 0.05$ or 0.2 m/s) and two camera pixel measurement noise levels ($\sigma_c = 0.1$ or 0.4 pixels). The columns are the error distribution plots for a calibration parameter.

tively. This sensitivity indicates that, prior to use in our algorithm, the radar data should be filtered to remove high-noise measurements whenever possible.

If the system follows the high-velocity trajectory in Figure 5.5b, then radar data filtering may not be necessary. As shown in Figure 5.7, calibrating the radar along this trajectory results in similar scale and rotation estimation accuracy as the low velocity trajectory, but drastically improves the translation and temporal offset estimates; the errors are within 10 cm and 10 ms, respectively. Additionally, our algorithm achieves a comparable spatial calibration error to Doer and Trommer [2020a] on noisier radar ego-velocity data. However, this high rotational velocity trajectory is challenging for real-world camera localization and is not necessary if the radar data are sufficiently accurate.

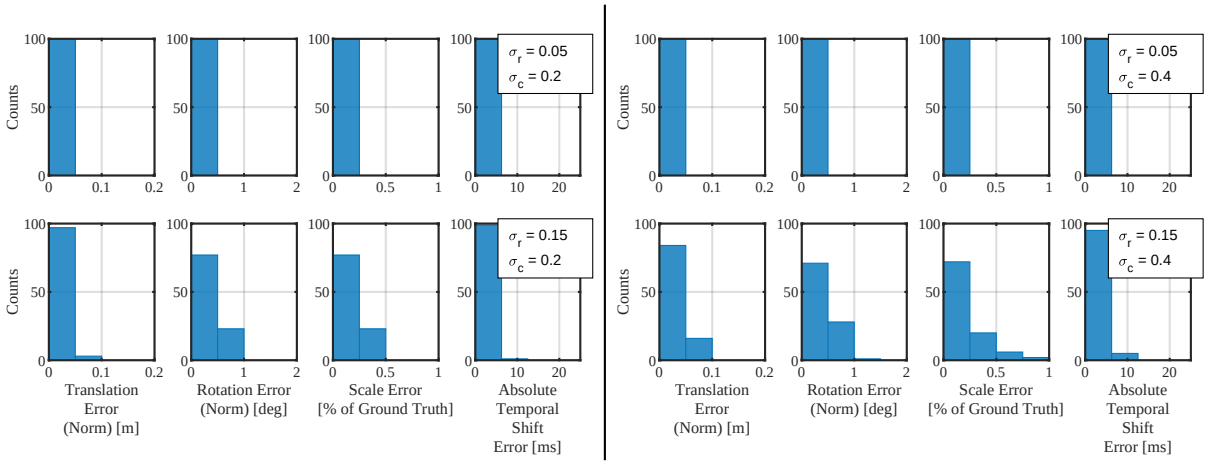


Figure 5.7: Low linear and high rotational velocity trajectory spatiotemporal calibration results from the simulated experiments. Each subplot is a histogram of the error between the estimated and true parameter values for 100 experiments at a given level of measurement noise. Each row presents the results for a level of measurement noise. The levels of measurement noise are a combination of two radar measurement noise levels ($\sigma_r = 0.05$ or 0.2 m/s) and two camera pixel measurement noise levels ($\sigma_c = 0.1$ or 0.4 pixels). The columns are the error distribution plots for a calibration parameter.

5.6.2 Real-World Experiments

To verify the performance and accuracy of our algorithm, we carried out a series of real-world experiments involving four different radar-camera systems. Initially, we discuss the various systems and their implementation details. Then, we show that radar-camera extrinsic calibration using ego-velocity measurements with known scale and temporal offset is possible. Next, we show that the set of spatiotemporal calibration parameters estimated by our algorithm have a similar level of alignment accuracy as the parameters estimated by the target-based method of Peršić et al. [2021b]. After comparing our spatiotemporal calibration parameters to Peršić et al. [2021b], we demonstrate how spatiotemporal calibration can improve the performance of camera-radar-IMU odometry. Finally, we evaluate the accuracy of our algorithm in a challenging situation involving sensors mounted on an autonomous vehicle.

Data Collection and Data Preprocessing

The data collection systems are different for each experiment, however each system consists of at least one radar and one camera. In the experiment with known scale and temporal offset, our data collection rig has a Texas Instruments (TI) AWR1843BOOST 3D radar and Point Grey BFLY-U3-23S6M-C camera. Both sensors have a measurement update rate of approximately 10 Hz. For the experiment that compares our method to the method in Peršić et al. [2021b], the system is a handheld rig that incorporates a TI AWR1843BOOST radar and Point Grey Flea3 camera. The measurement update rates for the sensors are 20 Hz and 30 Hz, respectively. For the experiment that involves the dataset from Doer and Trommer [2021], the data collection system is a handheld rig that mounts on a drone, where measurements are acquired from a TI IWR6843AOP unit, an IDS UI-3241 camera, and an Analog Devices ADIS16448 inertial measurement unit (IMU), operating at frequencies of 10 Hz, 20 Hz, and 409 Hz, respectively. Doer and Trommer [2021] provides additional details about this system. In the vehicle experiment, we use the data collection system from Burnett et al. [2023]. This system incorporates a vehicle-mounted TI AWR1843BOOST radar and three Point Grey Flea3 GigE cameras operating at 25 Hz and 16 Hz, respectively.

In our real-world experiments, we use two similar radars that primarily differ in azimuth and elevation resolutions. If two targets have an identical range and range-rate, and are separated by less than the azimuth or elevation resolution, then the targets will blend together which biases the radar measurement. The AWR1843BOOST has azimuth and elevation resolutions of 15° and 58°, respectively, while the IWR6843AOP has azimuth and elevation resolutions of 30°. As we show in this section, our algorithm is capable of calibrating both radars even though they have differing radar resolutions. For additional information on the radars used in our experiments, we refer the reader to Texas Instruments [2020, 2022], which are the user manuals for each radar.

To ensure accurate ego-velocity estimation for our experiments, we set the maximum measurable range-rate and constant false alarm rate (CFAR) thresholds for our radar units. The maximum range-rate of the radar must be set above the maximum velocity of the data collection platform because the ego-velocity estimates will saturate at this value. As shown in Richards et al. [2010], an inverse relationship exists between the maximum range-rate and maximum range settings and these must be properly balanced for the operating environment. The on-board radar pre-processing pipeline incorporates a CFAR detector that differentiates targets from background noise in the received EM signal. Readers searching for more information on CFAR detection should consult Richards et al. [2010]. Since the definition of background noise is also environment-dependant, we set the CFAR threshold to ensure that the ego-velocity estimator returned a sufficient number of inliers, while minimizing the number of outliers. Before each experiment, we performed a series of 'test' data collection runs

to tune these settings, ensuring that the ego-velocity estimates were not saturating, that there were at least 15 inliers for each measurement, and that the inlier to outlier ratio was above 50%.

There are three data preprocessing steps for the all experiments discussed in this section, while the vehicle experiment requires a fourth additional preprocessing step. Prior to estimating the calibration parameters using our algorithm, we first determine radar ego-velocity estimates using the algorithm from Doer and Trommer [2020b] or Stahoviak [2019].[2] Second, we rectify the camera images to remove lens distortion effects. Third, we use the feature-based, monocular simultaneous localization and mapping (SLAM) algorithm ORB-SLAM3 from Campos et al. [2021] to provide an initial estimate of the (arbitrarily-scaled) pose of the camera at the time of each image acquisition. While camera pose estimation is possible with any monocular SLAM, we chose this package for its robustness and accuracy. Finally, for the vehicle experiment, we remove outlier radar ego-velocity and camera pose estimates using a median filter. The median filter computes the local median and standard deviation of the signals across a window of time—200 ms and 850 ms for the radar and camera, respectively. If the measurement at the center of the window is greater than a chosen threshold from the median, the measurement is treated an outlier. For the runs in the vehicle experiment, the threshold is set to three standard deviations from the median, since this value eliminates gross outliers without removing noisy, but valid, portions of the signals. We found that this step was necessary to ensure data integrity.

Extrinsic Calibration with Known Scale and Temporal Offset

To evaluate the calibration accuracy of our algorithm with a known scale and temporal offset, we perform extrinsic calibration in an environment with a checkerboard target of known size and with the temporal offset estimate from the method in Peršić et al. [2021b]. Figure 5.8 shows the handheld rig and experimental environment. We compare the performance of our algorithm to the 3D reprojection-based algorithm of Peršić et al. [2021b]. Additionally, the extrinsic calibration (translation and rotation) parameters were carefully measured by hand for comparison.

Experiments were conducted outdoors to mitigate (to some extent) radar multipath reflections and other detrimental effects. We placed five specialized hybrid radar-camera targets in the environment for validation purposes and for comparison with the calibration method in Peršić et al. [2021b]. The trihedral retroreflective targets are specially-constructed for radar-camera calibration, and consist of a trihedral radar retroreflective 'corner' and a visual AprilTag pattern from Olson [2011] printed on paper. As shown in Figure 5.9, a target has the AprilTag (which is EM transparent) mounted in front of the retroreflector. Using the known AprilTag scale, the pose of the camera relative to the AprilTag reference frame can be established. The distance from the origin of the AprilTag frame to the corner of the retroreflector is also known, so the AprilTag enables the camera to observe the retroreflector corner. During data collection, we moved the radar-camera pair and kept the radar pointed at the reflector opening of at least one target to ensure a consistent radar reflection. However, we emphasize that our algorithm does not specifically make use of the retroreflective radar targets; the velocity of the radar can be determined independently. To enable full pose estimation using monocular camera data, the environment contains a checkerboard of known scale, as shown in Figure 5.8.

We evaluated the performance of the calibration algorithm by measuring target reprojection error. Using the extrinsic transform obtained via a given calibration method, the radar measurement of the target can be projected into the camera reference frame. The distance between the observed 3D position of the retroreflector corner (from image data) and the projected radar estimate of the retroreflector corner is the target reprojection error. The retroreflector is more consistently detected than the AprilTag, so we linearly interpolate the mea-

[2] Available at: https://github.com/christopherdoer/reve and https://github.com/cstahoviak/goggles

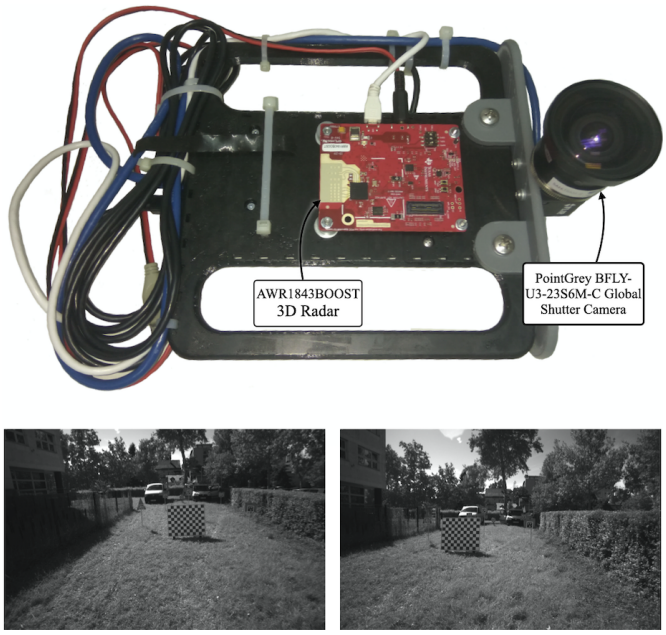


Figure 5.8: The top image is a picture of the handheld data collection rig. The bottom two images show different perspectives of our data collection environment.

sured position of the trihedral retroreflector corner by the radar at the image timestamps. Figure 5.10 shows the radar-to-camera reprojection error determined using three different calibration methods: hand-measurement, the 3D reprojection-based method of Peršić et al. [2021b], and our proposed method. Since the transform estimated by the 3D reprojection method in Peršić et al. [2021b] optimally aligns the AprilTag positions with the projected radar measurements of the targets, this approach outperforms our algorithm according to this metric, as expected. However, the difference in the median reprojection error between our proposed method and that in Peršić et al. [2021b] is less than 4 mm. In contrast to Peršić et al. [2021b], our algorithm does not require any specialized radar targets in the general case.

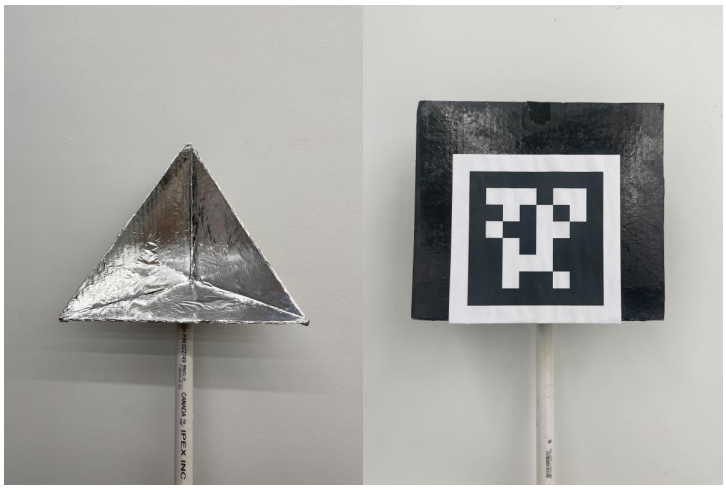


Figure 5.9: Our specialized retroreflective radar target used for verification. The left image shows the retroreflector alone, while the right image shows an AprilTag mounted to a flat cardboard backing that is attached to the front of the retroreflector. The cardboard material is fully transparent to the radar EM wave.

Handheld Rig Experiment

In this experiment, we compared the spatiotemporal calibration parameters estimated by our algorithm against the parameters determined by the target-based method in Peršić et al. [2021b]. To compare the two approaches, we used a handheld rig to collect a dataset consisting of two parts: one part with no visible calibration targets (for our algorithm) and one part with visible targets for target-based calibration. We collected both parts on the dataset during one continuous run, without power-cycling the sensors. Similar to the experiment with known scale and time offset, our quality metric is the reprojection error, so the target-based calibration can be treated as the 'gold standard,' effectively.

We used the first part of the dataset to perform targetless radar-camera calibration with our algorithm. The procedure consisted of moving the sensor rig, shown in Figure 5.11, throughout the office environment shown in Figure 5.12. A segment of the system trajectory estimated by our algorithm is plotted in Figure 2.2. Then, we used the second part of the dataset to perform target-based calibration with the algorithm described in Peršić et al. [2021b]. In this case, the procedure consisted of moving a trihedral retroreflective target, shown in Figure 5.9, in front of the stationary radar-camera rig. Similar to the known scale and temporal offset experiment, the target opening was directed at the radar to ensure consistent reflections and the measurements of the trihedral corner by the radar were linearly interpolated to the image timestamps. The second part of the dataset was also used to evaluate the relative accuracy of the parameters estimated by both algorithms.

Overall, our algorithm achieves a reprojection error distribution that is comparable to the method from Peršić et al. [2021b]. Table 5.2 shows that the estimated translation and rotation, are, per axis, within 1.6 cm and 3 degrees, respectively, of the values estimated by the target-based method, which is expected from the known scale and fixed temporal offset case. Additionally, our estimated temporal offset differs from the target-based method by only 6 ms. Figure 5.13 shows that our algorithm, in a completely targetless manner, produces a reprojection error distribution with a median that is only 3 mm larger than the target-based method.

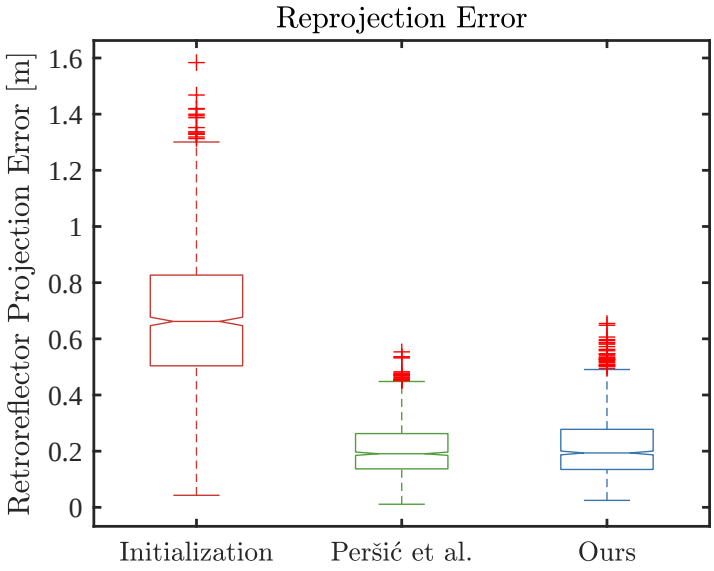


Figure 5.10: The target reprojection error is shown for the following calibration methods: hand-measured, Peršić et al. [2021b], and our proposed method. All algorithms used the same dataset and all calibration results were obtained from a held-out dataset.

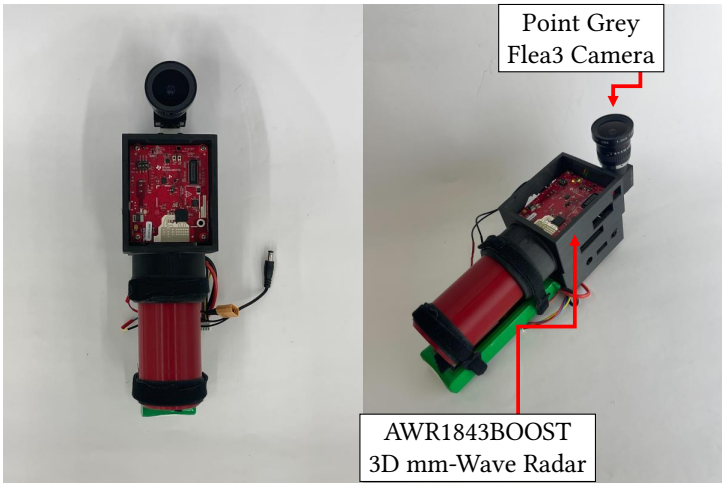


Figure 5.11: Two pictures of our handheld data collection rig. The left image is a front view and the right image is an isometric view of the radar-camera unit. The radar antennas are mounted in the white area on the red circuit board. From our CAD model of the handheld rig, the radar-camera translation parameters (i.e., the components of $\mathbf{t}_c^{rc}$) are $r_x = 0.1$, $r_y = 10.5$, and $r_z = -1.0$ cm.

IRS Radar Thermal Visual Inertial Datasets

In this section, we demonstrate the versatility of our algorithm by making use of our estimated calibration parameters to improve the accuracy of camera-radar-IMU odometry for the system described by Doer and Trommer [2021]. The extrinsic calibration parameters that accompany the IRS dataset were determined using the radar-IMU extrinsic calibration process described in Doer and Trommer [2020a] with ad-hoc temporal calibration. Post-hoc calibration of the radar and camera for the IRS is challenging because the test environments do not contain any trihedral reflectors (thus, target-based spatiotemporal calibration is not possible) and the motion of the sensor platform is constrained (i.e., there are no deliberate excitations for calibration). To the best of the authors' knowledge, our approach is the only technique that can estimate all of the spatiotemporal calibration parameters for the datasets described in Doer and Trommer [2021].

We chose to calibrate, and to evaluate the calibration quality, for three IRS datasets from the total of nine available datasets. We calibrated using the Gym, `MoCap Easy`, and `MoCap Medium` datasets. These datasets were collected in two environments with varying numbers of features: a large, sparse gymnasium and an feature-rich office. For the other datasets, poor lighting conditions and rapid motions caused ORB-SLAM3 to fail. To evaluate on a given dataset, we compute the radar-camera spatiotemporal calibration parameters using our algorithm, and then run RRxIO on the same dataset with our estimated parameters. During evaluation, we disable the live 'camera-to-IMU' extrinsic calibration algorithm that operates as part of in RRxIO. Using the known ground truth and the estimated RRxIO trajectories, we are able to determine the quality of our calibration using the following odometry error metrics: the relative translational root mean square error (RMSE RTE), relative rotational RMSE (RRE), absolute translational RMSE (ATE), and absolute rotational RMSE (ARE).

Table 5.2: Calibration parameters for our handheld dataset. The values in each row are estimated by a different algorithm. The rotation between the sensors is given in roll-pitch-yaw (i.e., θ_x, θ_y, θ_z) Euler angle form.

	r_x [cm]	r_y [cm]	r_z [cm]	θ_x [rads]	θ_y [rads]	θ_z [rads]	τ [ms]
Peršić et al. [2021b]	-1.60	11.9	-5.02	-1.59	0.07	-3.12	-63.8
Ours	-0.48	12.2	-3.42	-1.62	0.02	-3.15	-57.9

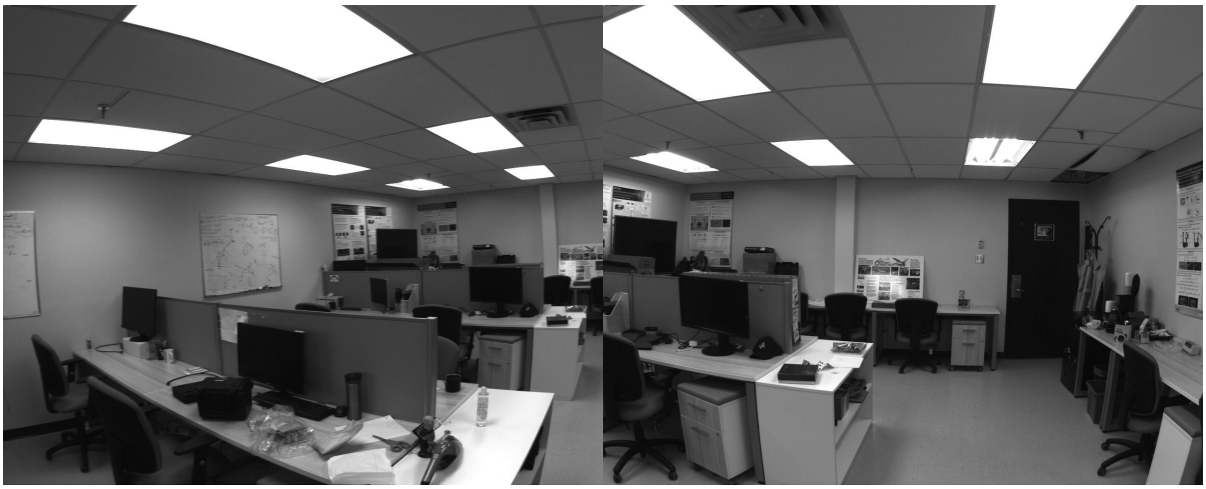


Figure 5.12: Images from our handheld sensor rig calibration dataset, showing two views of the feature-rich indoor test environment.

While the parameters estimated by our algorithm are relatively close to the parameters provided in the RRxIO paper, the use of our parameters result in more accurate odometry estimates. The parameters that our algorithm recovered for each dataset are shown in Table 5.3. The estimated temporal offset for the Gym dataset is the only large deviation from the RRxIO-provided value, but, from our experience, the radar temporal offset can change significantly between system power cycles. Table 5.4 reports the absolute and relative translation and rotation errors for the RRxIO trajectories after a yaw alignment. The parameters estimated by our algorithm improve the translation error on all datasets and rotation error for two of the datasets. Notably, the Gym dataset, which has the largest temporal offset, improves the most.

Vehicle Experiments

In this section, we verify the accuracy of our calibration algorithm by estimating the distance between cameras mounted on the autonomous vehicle presented in Burnett et al. [2023]. This task was challenging because, as shown in Figure 5.15, the radar-camera pairs do not share overlapping fields of view, so it is impossible to perform calibration using a target-based method. Additionally, the constrained motion of the car results in

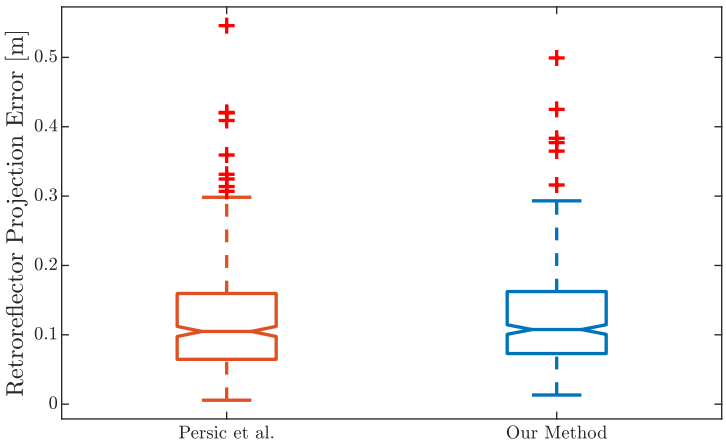


Figure 5.13: The reprojection error distributions for the state-of-the-art method in Peršić et al. [2021b] and ours.

Table 5.3: Radar-IMU calibration parameters evaluated using three IRS datasets and RRxIO. The radar-IMU calibration parameters estimated by our algorithm are a combination of the IRS IMU-camera parameters and the camera-radar parameters we determined. The rotation between the two sensors is given in roll-pitch-yaw (i.e., θ_x, θ_y, θ_z) Euler angle form.

	r_x [cm]	r_y [cm]	r_z [cm]	θ_x [rads]	θ_y [rads]	θ_z [rads]	τ [ms]
RRxIO	6.00	4.00	-4.00	-3.14	0.02	-1.59	8.00
ME† (ours)	4.08	4.71	-5.05	-3.12	0.01	-1.59	13.1
MM† (ours)	3.90	4.46	-5.63	-3.11	0.01	-1.59	15.4
Gym (ours)	3.27	4.48	-3.62	-3.15	-0.06	-1.60	40.7

† These datasets are MoCap Easy (ME) and MoCap Medium (MM).

Table 5.4: Performance evaluation for RRxIO and our algorithm on three IRS datasets. The metrics evaluated are the relative translational RMSE (RTE), relative rotational RMSE (RRE), absolute translational RMSE (ATE), and absolute rotational RMSE (ARE).

	RTE [%]		**RRE [deg/m]**		**ATE [m]**		**ARE [deg]**	
Dataset	RRxIO	Ours	RRxIO	Ours	RRxIO	Ours	RRxIO	Ours
ME†	0.809	0.669	0.084	0.089	0.177	0.144	1.567	1.918
MM†	1.377	1.097	0.122	0.095	0.351	0.260	2.522	2.027
Gym	1.170	0.752	0.076	0.054	0.308	0.195	2.087	1.349

† These datasets are MoCap easy (ME) and MoCap Medium (MM).

a poorly conditioned problem (i.e., the minimum eigenvalue of the identifiability matrix in Equation (5.56) is close to zero). The poor conditioning of the problem makes the estimated parameters very sensitive to sensor measurement noise, which can lead to inaccurate results. To overcome the poor conditioning of this system, we add an extrinsic calibration prior,

$$\begin{aligned} \mathbf{e}_{prior} &= \log(\mathbf{T}_{cr}^{-1}\mathbf{T}_{cr,prior}), \\ J_{prior} &= \mathbf{e}_{prior}{}^{\intercal}\boldsymbol{\Sigma}_{prior}^{-1}\mathbf{e}_{prior}, \\ \boldsymbol{\Sigma}_{prior} &= \begin{bmatrix} \sigma_t^2\,\mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \sigma_\theta^2\,\mathbf{I}_{3\times3} \end{bmatrix}, \end{aligned} \tag{5.59}$$

to the optimization problem. For our experiments, the prior for the extrinsic calibration parameters ($\mathbf{T}_{cr,prior}$) is derived from hand measurement. We set the prior uncertainty for the translation (σ_t) to 0.1 m along each axis, and the prior uncertainty for the rotation to (σ_θ) 30 degrees. The addition of this term stabilizes the estimation of the vertical translation between the radar and cameras, in particular. After optimization, less than 1% of the final cost value is due to the prior error term.

The mounting positions of the radar and three cameras on the car are shown in Figure 5.14, and the corresponding fields of view are shown in Figure 5.15. The the first camera is positioned at the centre of the car and faces in the direction of travel. The two other cameras are placed to the left and to the right of the centre camera and point roughly 45 degrees from left and right from the forward axis. The 3D radar is more than one metre away from all of the cameras, facing towards the rear of the car, opposite the direction of travel.

We collected a total of nine datasets from the radar and the cameras (three datasets per camera) while driving two laps of a figure eight pattern. The laps occurred in a sparse parking lot environment, where the radar and camera features lie at a substantial distance from the vehicle. We evaluated the accuracy of our estimated parameters by comparing the estimated distances between the centre camera and the two side cameras to the distances measured using a Leica Nova MS50 MultiStation. This method of comparison was selected in part because camera-to-camera extrinsic calibration is difficult for camera pairs that have minimal field of view overlap. Additionally, structural components of the car prevent direct measurement of the distance between the radar

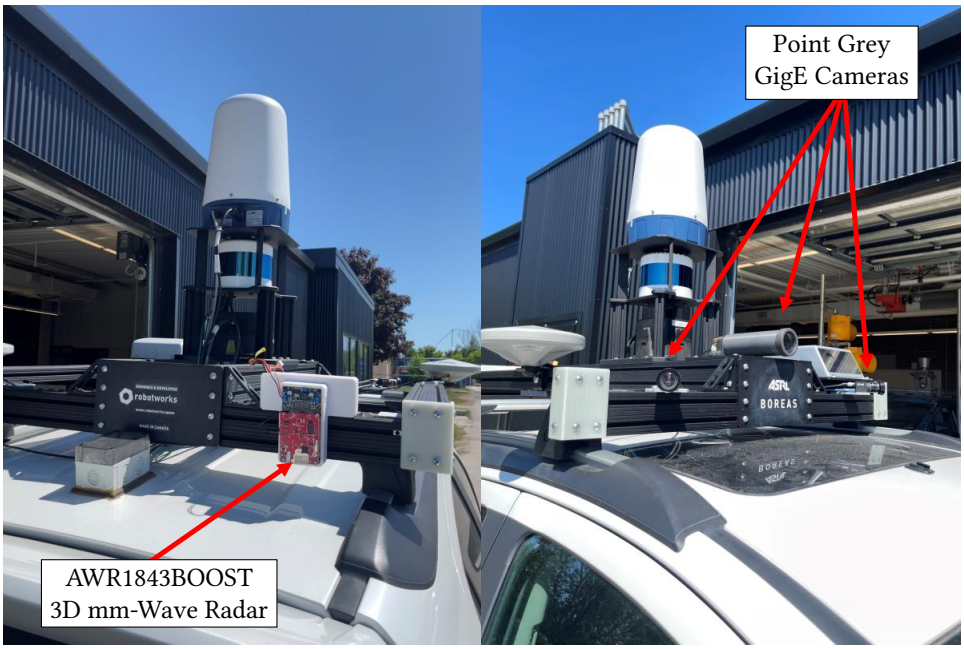


Figure 5.14: Two views of the radar and camera mounting positions on the vehicle used in our experiments. The left image shows the mounting position of the TI radar. The right image shows the mounting positions of the three Point Grey cameras. The radar and the cameras do not share overlapping fields of view.

and cameras. Each run of our spatiotemporal calibration algorithm produced an estimated extrinsic calibration, for a total of three sets of estimated extrinsic calibration parameters for each camera. The transformations between the centre-to-left and -right cameras are computed by combining two radar extrinsic calibration estimates, which give a total of 18 camera-to-camera extrinsic calibration estimates (nine left and nine right).

Figure 5.16 shows the distribution of distance errors. The majority of estimated extrinsic calibration parameters result in a camera-to-camera distance error of less than 5 cm, with two values that are greater than 10 cm. This is reasonable because we are chaining together two transforms, with translation magnitudes greater than 1 m, to confirm that the distance from the centre-to-left and -right cameras are both 35 cm.

Calibration Environment

Several notes are in order regarding environments that are suitable for calibration. Although our algorithm does not require any retroreflective targets for the radar or a specific calibration pattern for the camera, there are nonetheless some limitations on where calibration can be performed. To ensure accurate ego-velocity estimation, the calibration environment should contain, at minimum, four stationary features for ego-velocity estimation. Empirically, we have found that accurate ego-velocity estimation occurs when there are more than 10 detected features. To ensure accurate camera pose estimation using ORB-SLAM3, the scene should have sufficient lighting and visual texture. As a result, calibration should not be performed in scenes with many moving targets, dim lighting, or inclement weather (e.g., fog). However, the accuracy of the camera pose estimates are dependent on the SLAM algorithm chosen, which may or may not be robust to the previously mentioned situations.

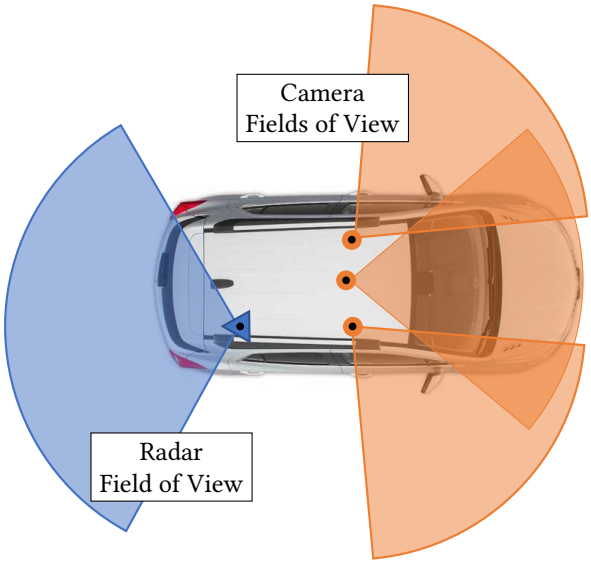


Figure 5.15: Fields of view of the radar and cameras sensors used for our vehicle experiments.

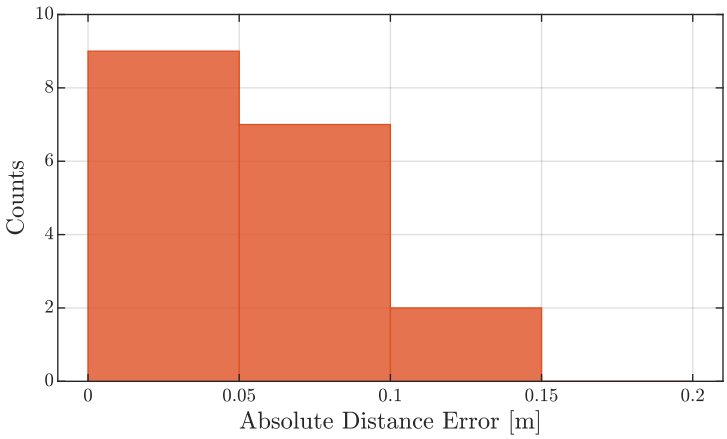


Figure 5.16: Results from the vehicle calibration experiment, where the radar and the cameras do not share overlapping fields of view. The distance error is the difference between the estimated and measured distances between the center-left and center-right cameras. The ground truth distance was determined using a Leica MultiStation.

5.7 Summary and Future Work

In this chapter, we presented a novel targetless radar-camera spatiotemporal calibration algorithm that leverages ego-velocity measurements as an abstraction of a 3D mm-wavelength radar unit. We proved that the calibration problem is identifiable and determined the necessary conditions for successful calibration. Through simulation studies, we demonstrated that our algorithm is accurate, but can be sensitive to the amount of noise present in the radar range-rate measurements. Further, we evaluated our algorithm in three different, real-world environments. First, we showed, using data from a handheld sensor rig, that our approach can match the accuracy of target-based calibration methods. Second, we presented results indicating that calibration can improve the localization performance of a hardware-triggered radar-camera-IMU system. Finally, we established that our calibration framework can be applied to AV systems, where the radar and camera are mounted at a significant distance from each other and do not share overlapping fields of view.

There are multiple potential directions for future research. Notably, we leave the complete characterization of the sets of unidentifiable trajectories as future work. Additionally, we could investigate alternative cost functions that explicitly consider alignment errors (similar to Peršić et al. [2021b]) and ego-velocity estimates. A valuable addition to our approach would be a method to automatically determine the knot spacing required for the continuous-time spline representation. Alternatively, as demonstrated by Barfoot [2024], we can use Gaussian processes for our continuous-time representation, which would eliminate the knot spacing issue. Further, the method could be naturally extended to the multi-camera and multi-radar setting. Finally, other pairs of sensors could also be considered beyond radar-camera pairs, including radar-inertial sensor combinations, for example.

Chapter 6

Extrinsic Calibration of 2D Radar Pairs

In this chapter, we investigate 2D radar-radar extrinsic calibration using radar ego-velocity estimates only. Although 2D radar data lacks the elevation information contained in 3D radar sensor data, we can examine the problem of estimating the $\mathrm{SE}(2)$ transform between 2D radar sensors An illustration of the problem setup is shown in Figure 6.1. This task provides a setting to answer the question: which calibration parameters can we estimate reliably using ego-velocity measurements from a pair of radar units that are limited to scanning in a horizontal plane?

We show in the sections below that we can recover the yaw angle and the direction of the translation vector between pairs of coplanar 2D radar sensors from ego-velocity estimates alone. While we are unable to estimate the *distance* between the sensors, this distance is unlikely to change appreciably from specifications under nominal operating conditions. In contrast, the orientation (yaw) of each radar can easily be altered (e.g., by a minor impact) and is difficult to measure by hand. Section 6.1 briefly reviews existing extrinsic calibration algorithms where one sensor is a 2D radar and where the $\mathrm{SE}(2)$ transform between the sensors is determined. Section 6.2 presents a batch solver for the 2D radar-pair extrinsic calibration problem. In Section 6.3, we prove that the extrinsic calibration parameters are identifiable, given sufficient excitation. Section 6.4 presents a series of simulation studies and real-world experiments that analyze the sensitivity of our method to varying levels of measurement noise. The results from these studies and experiments confirm that our approach is more accurate and reliable than two state-of-the-art methods. Further, we demonstrate that the full spatial transform can be recovered when an additional, coarse source of rotational velocity information is available.

6.1 Related Work

While Section 5.1 reviewed some prior work on extrinsic calibration involving 2D radars, we briefly revisit three extrinsic calibration methods that involve at least one 2D mm-wavelength radar unit and that estimate the $\mathrm{SE}(2)$ spatial transform between the sensors. The method in Olutomilayo et al. [2021] estimates the transformation that best aligns the point cloud measurements from two rigidly-connected 2D mm-wavelength radars. This method relies on mutually-observed trihedral reflectors. Burnett et al. [2023] describes an approach to estimate the $\mathrm{SE}(2)$ transform between a 2D radar and a laser ranging unit (lidar), without the need for trihedral reflectors. Specifically, this method applies scan matching to align radar and lidar point clouds, but does require a large number of jointly-observed features. Notably, co-observing a sufficient number of features is not always possible for radars and lidars with narrow fields of view. Finally, the method in Kellner et al. [2015] estimates the

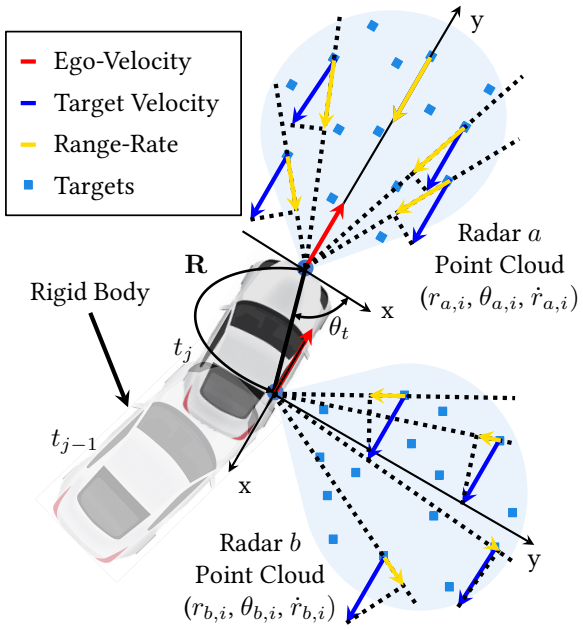


Figure 6.1: Illustration of our 2D radar-to-radar extrinsic calibration problem. Radars a and b measure the range, azimuth, and range-rate to targets in the environment. Assuming that the targets are stationary in a fixed, world reference frame, we can estimate the ego-velocity of each radar. Our algorithm fuses the ego-velocity estimates from both radars to estimate the rotation, $\mathbf{R}$, and translation direction θ_t between the sensors.

rotation between a car-mounted 2D radar and an IMU by minimizing the difference in estimated lateral velocities, expressed in the radar frame. While the radar ego-velocity measurements provide lateral velocity directly, determining the lateral velocity of the radar from IMU measurements requires the IMU rotational velocity and accurate knowledge of the radar-IMU translation. For each of these methods, 2D radar calibration is limited to environments that contain trihedral reflectors or to systems that have specific sensor configurations.

6.2 Problem Formulation

Let $\underrightarrow{\mathcal{F}}_a$ and $\underrightarrow{\mathcal{F}}_b$ be the reference frames of two rigidly-attached radar units that share and move in one horizontal sensing plane. Additionally, let $\underrightarrow{\mathcal{F}}_w$ be a fixed, inertial world frame. The rotation and translation between the sensors are elements of $\mathrm{SO}(2)$ and $\mathbb{R}^2$, respectively, and the ego-velocity measurement models for radars a and b, at time t_j, are

$$\mathbf{h}_a(t_j) = \mathbf{v}_a(t_j) + \mathbf{n}_a(t_j), \tag{6.1}$$

$$\mathbf{n}_a(t_j) \sim \mathcal{N}\left(\mathbf{0}, \mathbf{\Sigma}_a(t_j)\right) \tag{6.2}$$

$$\mathbf{h}_b(t_j) = \exp\left(\theta_{ba}{}^{\wedge}\right) \begin{bmatrix} 1 & 0 & -\mathbf{t}_{a,y}^{ba} \\ 0 & 1 & \mathbf{t}_{a,x}^{ba} \end{bmatrix} \begin{bmatrix} \mathbf{v}_a(t_j) \\ \omega(t_j) \end{bmatrix} + \mathbf{n}_b(t_j), \tag{6.3}$$

$$\mathbf{n}_b(t_j) \sim \mathcal{N}\left(\mathbf{0}, \mathbf{\Sigma}_a(t_j)\right) \tag{6.4}$$

where $\mathbf{h}_{r,a}\left(t_j\right)$ and $\mathbf{h}_{r,b}\left(t_j\right)$ are the estimated values from Equation (5.6) for radar measurements a and b at time t_j, respectively. The value $\mathbf{v}_a$ is the ego-velocity of radar a, θ_{ba} is the rotation from radar a to radar b, $\omega(t_j)$ is the rotational velocity of the rigid body relative to $\underrightarrow{\mathcal{F}}_w$, and $\mathbf{t}_a^{ba}$ is the translation from radar a to radar

b, expressed in the reference frame of a. The vectors $\mathbf{n}_a(t_j)$ and $\mathbf{n}_b(t_j)$ are additive zero-mean Gaussian noise terms with covariances $\mathbf{\Sigma}_a(t_j)$ and $\mathbf{\Sigma}_b(t_j)$, respectively. The values of $\mathbf{\Sigma}_a(t_j)$ and $\mathbf{\Sigma}_b(t_j)$ are determined with use of Equation (5.7). The associated error distributions are

$$\begin{aligned}\mathbf{e}_{a,j} &= \mathbf{h}_a\left(t_j\right) - \mathbf{v}_a(t_j) \sim \mathcal{N}\left(\mathbf{0}, \mathbf{\Sigma}_a(t_j)\right), \\ \mathbf{e}_{b,j} &= \mathbf{h}_b\left(t_j\right) - \exp\left(\theta_{ba}{}^{\wedge}\right) \begin{bmatrix} 1 & 0 & -\mathbf{t}_{a,y}^{ba} \\ 0 & 1 & \mathbf{t}_{a,x}^{ba} \end{bmatrix} \begin{bmatrix} \mathbf{v}_a(t_j) \\ \omega(t_j) \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{\Sigma}_b(t_j)\right).\end{aligned} \tag{6.5}$$

The nonzero Jacobians of the error equations are

$$\frac{\partial \mathbf{e}_{a,j}}{\partial \mathbf{v}_a(t_j)} = -\mathbf{I}, \tag{6.6}$$

$$\frac{\partial \mathbf{e}_{b,j}}{\partial \mathbf{v}_a(t_j)} = -\exp\left(\theta_{ba}{}^{\wedge}\right), \tag{6.7}$$

$$\frac{\partial \mathbf{e}_{b,j}}{\partial \omega(t_j)} = -\exp\left(\theta_{ba}{}^{\wedge}\right) \begin{bmatrix} -\mathbf{t}_{a,y}^{ba} \\ \mathbf{t}_{a,x}^{ba} \end{bmatrix}, \tag{6.8}$$

$$\frac{\partial \mathbf{e}_{b,j}}{\partial \mathbf{t}_a^{ba}} = -\omega(t_j)\exp\left(\theta_{ba}{}^{\wedge}\right) \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}, \tag{6.9}$$

$$\frac{\partial \mathbf{e}_{b,j}}{\partial \theta_{ba}} = -\begin{bmatrix} -\sin\left(\theta_{ba}\right) & -\cos\left(\theta_{ba}\right) \\ \cos\left(\theta_{ba}\right) & -\sin\left(\theta_{ba}\right) \end{bmatrix} \begin{bmatrix} 1 & 0 & -\mathbf{t}_{a,y}^{ba} \\ 0 & 1 & \mathbf{t}_{a,x}^{ba} \end{bmatrix} \begin{bmatrix} \mathbf{v}_a(t_j) \\ \omega(t_j) \end{bmatrix}. \tag{6.10}$$

Given M pairs of synchronized radar measurements, the vector of parameters that we wish to estimate includes the ego-velocities of radar a and the rotational velocity of radar a with respect to $\underrightarrow{\mathcal{F}}_w$, expressed in $\underrightarrow{\mathcal{F}}_a$, from time t_1 to t_M. Additionally, the state vector contains the translation from radar a to b expressed in $\underrightarrow{\mathcal{F}}_a$, and the rotation angle from radar $\underrightarrow{\mathcal{F}}_a$ to $\underrightarrow{\mathcal{F}}_b$. As a result, the state vector takes the form

$$\mathbf{x}^{\intercal} = \begin{bmatrix} \mathbf{v}_a(t_1)^{\intercal} & \omega(t_1) & \cdots & \mathbf{v}_a(t_M)^{\intercal} & \omega(t_M) & \mathbf{t}_a^{ba\intercal} & \theta_{ba} \end{bmatrix}. \tag{6.11}$$

Our calibration problem is as follows.

Problem 11. *2D Radar-to-Radar Extrinsic Calibration Problem*

$$\min_{\mathbf{x}} \sum_{j=1}^{M} \mathbf{e}_{a,j}{}^{\intercal} \mathbf{\Sigma}_a(t_j)^{-1} \mathbf{e}_{a,j} + \mathbf{e}_{b,j}{}^{\intercal} \mathbf{\Sigma}_b(t_j)^{-1} \mathbf{e}_{b,j}. \tag{6.12}$$

6.3 Observability Analysis

In this section, we prove that, given sufficient excitation of the system, a subset of the parameters in Equation (6.11) are identifiable. Initially, we show that variations in the translation scale and rotational velocity are indistinguishable. To deal with this, we eliminate one degree of freedom by reducing the state space. We then prove that, for the reduced set of parameters, the problem is locally weakly observable. Since a problem that is locally weakly observable is also identifiable (in the batch setting), we use the rank criterion defined by Hermann and Krener [1977] in our proof. We then discuss a method to initialize the state vector. Finally, we highlight important degenerate motions which result in a loss of observability and potentially also of identifiability.

6.3.1 Scale and Rotational Velocity Indistinguishability

Unfortunately, Problem 11 has infinitely many indistinguishable solutions. Given any solution that minimizes Problem 11, another minimizer can be found by arbitrarily scaling $\omega(t_j)\ \forall\ j = 1, \ldots, M$ and $\mathbf{t}_a^{ba}$ by $\gamma \in \mathbb{R}$ and $\frac{1}{\gamma}$, respectively. However, the problem can be made distinguishable (identifiable) by reducing the size of the state space.

To make the optimization problem identifiable (see Section 6.3.2), we constrain

$$\left\|\mathbf{t}_a^{ba}\right\|_2 = 1. \tag{6.13}$$

We enforce this constraint by setting

$$\mathbf{t}_a^{ba} = \begin{bmatrix}\cos(\theta_t)\\ \sin(\theta_t)\end{bmatrix}, \tag{6.14}$$

where θ_t is the angle from the x-axis of radar a to the line of possible translations between radars a and b. By enforcing this constraint, the estimated rotational velocity is scaled by the magnitude of the translation between radars a and b. We denote the scaled rotational velocity as $\omega_\gamma(t_j)$. Additionally, a solution with θ_t and $\omega_\gamma(t_j)$ is indistinguishable from a solution with $\theta_t + \pi$ and $-\omega_\gamma(t_j)$, so we ensure $0 \leq \theta_t < \pi$. Consequently, our vector of parameters for the optimization problem becomes

$$\mathbf{x}^\intercal = \begin{bmatrix}\mathbf{v}_a(t_1)^\intercal & \omega_\gamma(t_1) & \cdots & \mathbf{v}_a(t_M)^\intercal & \omega_\gamma(t_M) & \theta_t & \theta_{ba}\end{bmatrix}. \tag{6.15}$$

We substitute Equation (6.14) into Problem 11 and solve the problem using the Levenberg-Marquardt algorithm.

6.3.2 Observability of Extrinsic Calibration

Let $\mathbf{h}_{r,a}(t_j)$ and $\mathbf{h}_{r,b}(t_j)$ be 2D radar ego-velocity measurements. We define the state at time t_j as

$$\mathbf{x}^T(t_j) = \begin{bmatrix}\omega_\gamma(t_j) & \alpha_\gamma(t_j) & \theta_t & \theta_{ba}\end{bmatrix}, \tag{6.16}$$

where $\alpha_\gamma(t_j)$ is the instantaneous rotational acceleration.[1] The motion model for the system is

$$\dot{\mathbf{x}}^T(t_j) = \begin{bmatrix}\alpha_\gamma(t_j) & \mathbf{0}_{3\times1}^T\end{bmatrix}. \tag{6.17}$$

We can substitute Equation (6.1) into Equation (6.3), which simplifies the measurement model to

$$\mathbf{h}_b(t_j) = \exp\left(\theta_{ba}{}^\wedge\right)\left(\omega_\gamma(t_j)\begin{bmatrix}-\sin\theta_t\\ \cos\theta_t\end{bmatrix} + \mathbf{h}_a(t_j)\right). \tag{6.18}$$

The zeroth-order Lie derivative is

$$\nabla_{\mathbf{x}} L^0\mathbf{h}_b(t_j) = \begin{bmatrix}\mathbf{a} & \mathbf{0} & \mathbf{b} & \mathbf{c}\end{bmatrix}, \tag{6.19}$$

$$\mathbf{a} = \exp\left(\theta_{ba}{}^\wedge\right)\begin{bmatrix}-\sin\theta_t\\ \cos\theta_t\end{bmatrix}, \tag{6.20}$$

[1] The analysis can, in fact, be simplified by removing the rotational acceleration state; we use this formulation, specifically, in Section 6.3.4.

$$\mathbf{b} = \omega_\gamma(t_j)\exp\left({\theta_{ba}}^\wedge\right)\begin{bmatrix}-\cos\theta_t\\-\sin\theta_t\end{bmatrix}, \tag{6.21}$$

$$\mathbf{c} = \begin{bmatrix}-\sin(\theta_{ba}) & -\cos(\theta_{ba})\\ \cos(\theta_{ba}) & -\sin(\theta_{ba})\end{bmatrix}\left(\omega_\gamma(t_j)\begin{bmatrix}-\sin\theta_t\\\cos\theta_t\end{bmatrix} + \mathbf{h}_a(t_j)\right). \tag{6.22}$$

The first-order Lie derivative is

$$\nabla_{\mathbf{x}} L_f^1 \mathbf{h}_b(t_j) = \begin{bmatrix}\mathbf{0} & \mathbf{d} & \mathbf{e} & \mathbf{g}\end{bmatrix}, \tag{6.23}$$

$$\mathbf{d} = \exp\left({\theta_{ba}}^\wedge\right)\begin{bmatrix}-\sin\theta_t\\\cos\theta_t\end{bmatrix}, \tag{6.24}$$

$$\mathbf{e} = \alpha_\gamma(t_j)\exp\left({\theta_{ba}}^\wedge\right)\begin{bmatrix}-\cos\theta_t\\-\sin\theta_t\end{bmatrix}, \tag{6.25}$$

$$\mathbf{g} = \alpha_\gamma(t_j)\begin{bmatrix}-\sin(\theta_{ba}) & -\cos(\theta_{ba})\\ \cos(\theta_{ba}) & -\sin(\theta_{ba})\end{bmatrix}\begin{bmatrix}-\sin\theta_t\\\cos\theta_t\end{bmatrix}. \tag{6.26}$$

The observability matrix of this system can be written as

$$\mathbf{O} = \begin{bmatrix}\nabla_{\mathbf{x}} L^0 \mathbf{h}_b(t_j)\\ \nabla_{\mathbf{x}} L_f^1 \mathbf{h}_b(t_j)\end{bmatrix}, \tag{6.27}$$

which has full column rank except when the sensor platform motion is degenerate, as discussed in Section 6.3.4.

6.3.3 Problem Initialization

Since measurements from the two radar sensors provide no rotational velocity information, we require a method to initialize the rotational velocities that appear in Equation (6.15). To begin, we determine θ_{ba} by finding the K pairs of ego-velocity estimates that have similar magnitudes. Using these 'velocity pairs,' we compute

$$\theta_{ba,l} = \arcsin\left(\begin{bmatrix}0 & 0 & 1\end{bmatrix}\left(\begin{bmatrix}\mathbf{h}_a(t_l)\\0\end{bmatrix}\times\begin{bmatrix}\mathbf{h}_b(t_l)\\0\end{bmatrix}\right)\right). \tag{6.28}$$

The initial value of θ_{ba} is the median of $\theta_{ba,l}\ \forall\ l = 1, ..., K$. To initialize θ_t, we use

$$\begin{aligned}\mathbf{b}_j &= \begin{bmatrix}b_{x,j} & b_{y,j}\end{bmatrix}^\mathsf{T} = \exp\left(-{\theta_{ba}}^\wedge\right)\mathbf{h}_{r,b}(t_j) - \mathbf{h}_{r,a}(t_j),\\ \theta_{t,j} &= \text{arctan2}\left(\frac{b_{y,j}}{\|\mathbf{b}_j\|_2}, -\frac{b_{x,j}}{\|\mathbf{b}_j\|_2}\right).\end{aligned} \tag{6.29}$$

Each $\theta_{t,j}$ is mapped to the corresponding value within $[0, \pi)$ and the initial θ_t is the median of $\theta_{t,j}\ \forall\ j = 1, ..., M$. By fixing θ_t and θ_{ba}, Problem 11 becomes an unconstrained quadratic problem. We use a standard linear solver to determine the initial estimates for $\omega_\gamma(t_j)$ and $\mathbf{v}_a(t_j)\ \forall\ j = 1 \ldots M$.

6.3.4 Degeneracy Analysis

The system is unobservable (and potentially unidentifiable) when $\mathbf{O}$ does not have full column rank. The determinant of the observability matrix is

$$\det(\mathbf{O}) = \begin{bmatrix} 0 & 0 & \alpha_\gamma(t_j) \end{bmatrix} \left(\begin{bmatrix} \mathbf{h}_a(t_j) \\ 0 \end{bmatrix} \times \begin{bmatrix} \cos\theta_t \\ \sin\theta_t \\ 0 \end{bmatrix} \right), \tag{6.30}$$

which is rank-deficient when $\det(\mathbf{O}) = 0$. As a result, the system must have nonzero rotational acceleration, α_γ, and nonzero ego-velocity, $\mathbf{h}_a(t_j)$. Additionally, the direction of ego-motion must not align with the sensor translation axis.

6.4 Experiments

To verify the performance of our algorithm, we conducted a series of simulated and real-world experiments. First, we show, using simulated data, that our algorithm is robust to realistic levels of radar measurement noise and that it yields an improved ego-velocity estimate. Second, we compare our approach to two state-of-the-art methods on the publicly-available Endeavour dataset.[2]

6.4.1 Simulation Studies

We performed a series of simulation studies to evaluate the robustness of our algorithm to measurement noise. We varied the simulation duration and the level of noise and generated 100 randomized trials with each pair of settings. Each simulation ranged in duration from 15 s to 120 s; the simulated sensor platform followed a periodic, nominal (noise-free) trajectory with sufficient excitation for our calibration problem (see Figure 6.2). The radar ego-velocity estimates for radars a and b were computed using the ground truth linear and rotational velocities of the platform along the trajectory. Ego-velocity measurements from radars a and b were then corrupted with zero-mean Gaussian noise ($\mathbf{\Sigma}_a(t_j) = \mathbf{\Sigma}_b(t_j) = \sigma_r^2\mathbf{I}_2$), where the standard deviation of the noise (σ_r) ranged from 0.05 m/s to 0.2 m/s. Based on our experiments (discussed in more detail in Section 6.4.2), we found the real-world measurement noise to be at the lower end of this range.

The error distribution of the estimated calibration parameters is shown in Figure 6.3. For most noise levels and durations, our estimated translation direction and rotation angle are, respectively, within 2° and 3° of the ground truth. Figure 6.4 shows that the median of the estimated ego-velocity errors for radars a and b are both 4 cm/s lower than the raw estimates. Importantly, this improvement can be achieved without the need for additional rotation information.

6.4.2 Real-World Experiments

We demonstrate the reliability of our method and compare to two state-of-the-art algorithms on the Endeavour dataset. Post-hoc extrinsic calibration for this dataset is challenging because the environments contain no trihedral reflectors. We demonstrate that the lack of trihedral reflectors has a negligible impact on our method, but is detrimental to the method in Olutomilayo et al. [2021]. Additionally, we show that the parameters estimated by our method result in smaller velocity errors than the parameters estimated by two state-of-the-art

[2]The dataset is available at: https://gloryhry.github.io/2021/06/25/Endeavour_Radar_Dataset.html

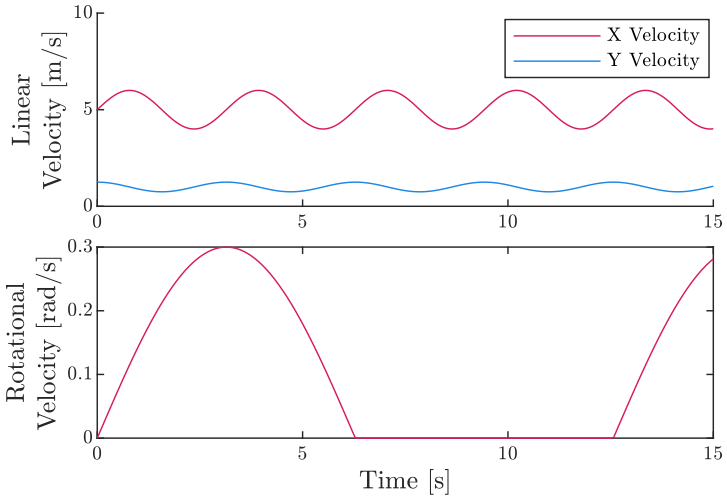


Figure 6.2: Top: ego-velocity of radar a over 15 s. Bottom: rotational velocity of radar a over 15 s. Both plots show the full period of the velocity functions.

methods. The first technique follows the approach in Olutomilayo et al. [2021]. To build the required map for this method, we collate measurements from one radar while the vehicle is stationary. The second method is similar to the approach in Burnett et al. [2023]. The parameters estimated by this method are included in the Endeavour dataset. Next, we demonstrate that normal driving motions provide sufficient excitation to calibrate radar pairs that have translation axes which align with the forward direction of the vehicle. Finally, we show that, when a source of rotational velocity information is available, the scale of the translation between the radar pair can be estimated.

The Endeavour dataset was collected from a small shuttle bus driving around three different loops in a campus setting. The dataset contains two runs for each loop, where each run is roughly 10 minutes long. The shuttle bus has a BDStar Navigation Npos320 RTK GNSS, four Velodyne VLP-16 lidars, and five Continental ARS430 radars, which operate at 100 Hz, 10 Hz, and 14 Hz, respectively. The radar labelled `Near_5` is mounted

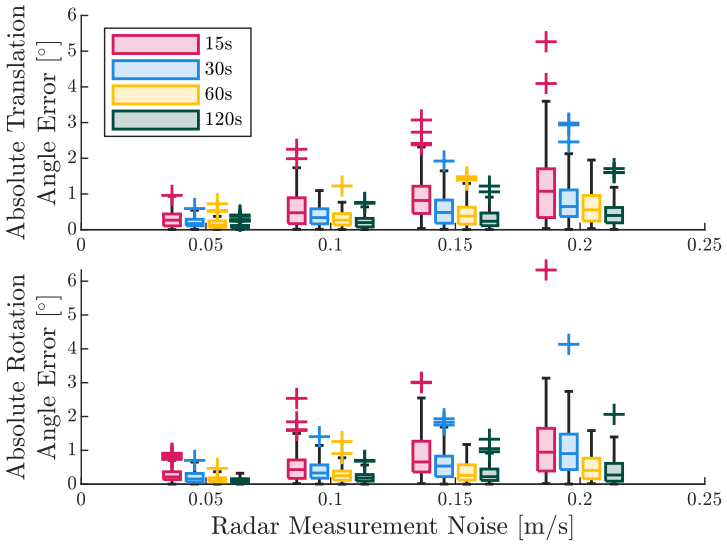


Figure 6.3: Absolute translation direction and rotation angle error for our algorithm at varying levels of measurement noise and simulation durations. The translation direction is the angle from the x-axis of radar a to the line of indistinguishable translations between radars a and b.

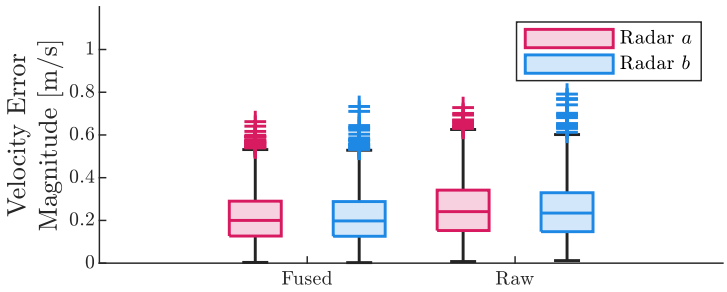


Figure 6.4: Raw and fused ego-velocity estimate errors of radars a and b for an experiment that is 120 s in duration with a measurement noise level of 0.2 m/s.

Table 6.1: θ_t and θ_{ba} parameters estimated by each method on `East2`. Radar a is `Near_5` for every parameter given.

	`Near_1`		`Near_2`		`Near_3`		`Near_4`	
Method	θ_t	θ_{ba}	θ_t	θ_{ba}	θ_t	θ_{ba}	θ_t	θ_{ba}
Endeavour	1.71	-1.57	1.44	1.59	2.73	-1.58	0.33	1.56
Olutomilayo et al. [2021]	1.36	-1.60	1.41	1.58	2.16	-1.61	0.39	1.56
Ours	1.74	-1.58	1.41	1.61	2.79	-1.58	0.35	1.56

*All angles are in radians.

on the front bumper and observes the environment in front of the vehicle. Radar pairs `Near_3`–`Near_1` and `Near_4`–`Near_2` observe the environment surrounding the sides of the vehicle. Radars `Near_3` and `Near_1` are mounted on the front and back driver's side of the vehicle, while radars `Near_4` and `Near_2` are mounted on the front and back passenger side of the vehicle.

Before applying our method to the Endeavour dataset, we tuned our RANSAC-based ego-velocity estimator, synchronized the radar measurement timestamps and removed zero-velocity measurement pairs. For RANSAC, the inlier and outlier thresholds were set to 40% of the number of measured reflections and 0.025 m/s, respectively. These thresholds were determined using the radar and GNSS velocity data from `East_2`. To temporally synchronize the data streams, we aligned the radar measurement timestamps using linear interpolation. Finally, we removed ego-velocity measurement pairs with magnitudes less than 0.05 m/s to improve the signal-to-noise ratio in the calibration problem.

Estimating the transforms for Olutomilayo et al. [2021] required two pre-processing steps. First, we identified stationary radar measurements using the RTK GNSS data and removed points that were observed less than five times. Next, we expressed (using the Endeavour parameters) the radar point clouds in a common reference frame and associated points that were within a 10 cm threshold. Finally, the extrinsic transforms from Olutomilayo et al. [2021] and Endeavour were chained together to compute the rotation angles and translation axes relative to `Near_5`.

Table 6.1 shows that our estimates are within 3° of the provided values, while the values from Olutomilayo et al. [2021] deviate significantly from the provided values. This deviation is due to the narrow overlap between the fields of view of some of the radar pairs, which results in sparse overlapping point clouds. Often, this systematic issue results in the data collection runs having insufficient information for the method in Olutomilayo et al. [2021] to operate properly (see Table 6.2).

We use the mean velocity error magnitude to evaluate the accuracy of the calibration parameters given in Table 6.1. The velocity error of a radar measurement pair, $\mathbf{h}_a(t_j)$ and $\mathbf{h}_b(t_j)$, is $\mathbf{e}_{b,j}$ from Equation (6.5), where $\mathbf{v}_a(t_j) = \mathbf{h}_a(t_j)$, θ_{ba} is the estimated rotation, θ_t is the estimated translation axis, and $\omega_\gamma(t_j)$ is the value that minimizes the magnitude of $\mathbf{e}_{b,j}$. Table 6.3 shows the mean velocity error magnitude for each run, radar pair,

and set of parameters. Our parameters yield lower velocity errors in almost all cases, reducing the mean error magnitude for the `Near_5`–`Near_4` radar pair by over 1 cm/s.

Due to the configuration of radar pairs `Near_3`–`Near_1` and `Near_4`–`Near_2`, the ego-motion of the vehicle driving forward aligns with the translation axes of these pairs, which, in theory, should make the calibration data poorly conditioned. However, these radar pairs are mounted on the periphery of the vehicle, so any rotational velocity induces unaligned ego-motion measurements. Carrying out calibration on data from pairs `Near_3`–`Near_1` and `Near_4`–`Near_2`, with initial calibration parameters greater than 20° from the Endeavour values, results in estimated parameters that are consistently within 3° of the Endeavour values; this indicates that the problem is not poorly conditioned.

By including a third sensor that is able to measure rotational velocity, we can estimate the (metric) scale of the translation between the radars, without requiring the exact extrinsic transform of the third sensor to be known. The magnitude of the rotational velocity of a rigid body is the same for all points on the body, allowing us to match the unscaled radar estimate to the rotational velocity of the third sensor. For example, assuming that the z-axis of an on-board GNSS receiver is roughly perpendicular to the sensing plane of the radar units, we can apply constant-acceleration smoothing and linear interpolation of the GNSS pose measurements to estimate rotational velocity. We tried this approach on the Endeavour dataset. After removing measurement pairs with rotational velocity magnitudes less than 0.1 rad/s, we computed the translation estimates. Table 6.4 shows that, in most cases, the metric translation values recovered by our algorithm are closer to the ground-truth Endeavour dataset values than those estimated by Olutomilayo et al. [2021]. While the sign of the translation may still be positive or negative (i.e., one z-axis could be inverted), this information can be easily determined from a rough model of the system.

6.5 Summary and Future Work

This chapter presented a 2D radar-to-radar extrinsic calibration algorithm that uses radar ego-velocity data only. Importantly, we have proved that the yaw angle and the axis of translation between the sensors can be identified given sufficient excitation. Using simulations, we demonstrated that our calibration method is robust to varying levels of radar measurement noise and that we are able to improve the raw radar ego-velocity estimates. Finally, we showed, using data from a vehicle, that our algorithm was more reliable and accurate than a state-of-the-art method.

There are multiple potential directions for future research. We could extend our approach to 3D radar sensors, similar to those discussed in Wise et al. [2021] and Wise et al. [2023]. We could also carry out temporal calibration using ego-velocity estimates, which may simplify the estimation problem for some systems. Finally, our calibration algorithm enables unbiased ego-velocity and biased rotational velocity estimates (i.e., the rota-

Table 6.2: Identifiability of Olutomilayo et al. [2021] for each run and radar pair in the Endeavour dataset. Our algorithm is identifiable for each run and radar pair.

	Data Collection Run					
Radar Pairs	East1	East2	Mid1	Mid2	West1	West2
Near_1–Near_3	✓	✓	✓	✓	✓	✓
Near_2–Near_4	✓	✓	✓	✓	✓	–
Near_3–Near_5	–	✓†	–	–	–	–
Near_4–Near_5	–	✓	–	–	–	–

† This problem is only identifiable using features that appear in less than 5% of measurements.

Table 6.3: Endeavour mean velocity error magnitude for each run excluding `East2`, which was used to estimate the calibration parameters. The errors presented below show how consistent the estimated parameters are when explaining the velocity vector field of a system with a unit moment arm.

	Near_5-Near_1			Near_5-Near_2			Near_5-Near_3			Near_5-Near_4		
Data	Endeavour	Olutomilayo	Ours	Endeavour	Olutomilayo	Ours	Endeavour	Olutomilayo	Ours	Endeavour	Olutomilayo	Ours
Mid1	0.0218	0.0591	**0.0175**	0.0232	0.0242	**0.0173**	0.0185	0.0928	**0.0173**	0.0411	0.0334	**0.0184**
Mid2	0.0215	0.0580	**0.0170**	0.0228	0.0235	**0.0166**	**0.0195**	0.0885	0.0288	0.0289	0.0240	**0.0140**
East1	0.0206	0.0657	**0.0152**	0.0208	0.0212	**0.0157**	0.0152	0.0846	**0.0139**	0.0305	0.0249	**0.0121**
West1	0.0206	0.0574	**0.0175**	0.0247	0.0249	**0.0179**	0.0151	0.1323	**0.0146**	0.0354	0.0285	**0.0149**
West2	0.0210	0.0558	**0.0176**	0.0259	0.0266	**0.0192**	0.0189	0.1320	**0.0178**	0.0515	0.0411	**0.0224**

*All values are in m/s.

Table 6.4: Estimated translation magnitude for each method on `East2`. Radar a is `Near_5` for every parameter given. The bolded values are the translation magnitudes closest to the Endeavour parameters.

Method	Near_1	Near_2	Near_3	Near_4
Endeavour	5.68	5.82	0.83	0.86
Olutomilayo et al. [2021]	4.87	**6.04**	0.47	0.96
Ours	**6.08**	5.54	**0.77**	**0.95**

*All values are in m.

tional velocity is scaled). Similar to Forster et al. [2017], we can integrate these velocities across time to form a biased discrete motion model, which would simplify radar data association between measurements at different times.

Chapter 7

Certifiable Hand-Eye Calibration

Usually, the term hand-eye (HE) calibration usually refers to the process of determining the transformation between a robotic manipulator and a pose sensor mounted on or held by the manipulator. However, Brookshire and Teller [2011, 2012] show that the common $\mathbf{AX} = \mathbf{XB}$ formulation of this problem can be applied to any sensor pair, when both sensors can be abstracted as pose sensors (e.g., stereo cameras, 3D lidar units, and GNSS-INS devices). Throughout this chapter, we refer to the sensor-agnostic $\mathbf{AX} = \mathbf{XB}$ problem as HE calibration, where $\mathbf{X}$ is the estimated transformation. Figure 7.1 is a diagram of the HE calibration formulation. While Giamou et al. [2019] present a certifiably optimal solver for the HE problem, their method does not support 'scaled' pose sensors (e.g., monocular cameras), which are frequently used in robotics.

We extend the certifiably optimal hand-eye calibration method in Giamou et al. [2019] to solve the monocular hand-eye calibration problem. Specifically, we extrinsically calibrate two rigidly-connected sensors, where one sensor is a pose sensor and the other sensor is a scaled pose sensor (i.e., can be abstracted as such). Going forward, we refer to each sensor as either a pose or scaled pose sensor for brevity. In Section 7.1 we survey traditional HE calibration methods. Starting from a probabilistic MLE problem, Section 7.2 extends the formulation in Giamou et al. [2019] to include a scaled pose sensor in the HE calibration problem. Section 7.3 presents a fast and certifiably optimal solver that leverages the Lagrangian dual of our problem. In Section 7.4, we use simulation studies to compare the estimation accuracy of our algorithm to another HE calibration solver.[2] Finally, we offer concluding remarks and ideas for future work in Section 7.5.

7.1 Related Work

The hand-eye (HE) calibration problem has been studied since the 1980s and we direct readers to the short literature reviews in Heller et al. [2014] and Hu et al. [2019] for more information on recent approaches. Herein, we review three categories of methods: separate, joint, and probabilistic. Table 7.1 briefly summarizes the characteristics of all the algorithms discussed in this section.

Early HE calibration algorithms solve for the rotation and translation parameters separately. Given the rotation parameters, estimating the translation is trivial, so these methods focus on fast, closed-form solutions for estimating the rotation between the sensors. As a result, the primary difference between these methods are the geometric constraints used in the rotation estimation problem. Tsai and Lenz [1989] develop a geometric

[2] See https://github.com/utiasSTARS/certifiable-calibration for Python code implementing our algorithm and experiments.

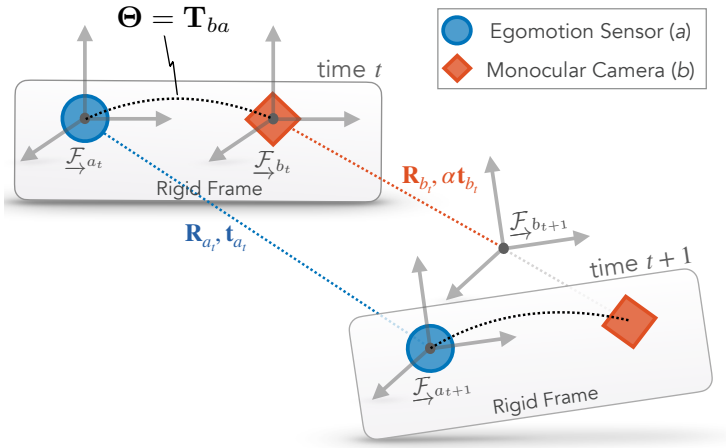


Figure 7.1: We perform extrinsic calibration between a sensor (a) that is able to provide egomotion estimates with known scale and a monocular camera (b) that cannot observe the scale, α, of its translational motion. A quadratically-constrained quadratic program (QCQP) formulation of hand-eye calibration is relaxed into a convex semidefinite program (SDP) which can be efficiently solved to yield a globally optimal estimate of the extrinsic transformation matrix $\mathbf{T}_{ba}$. The use of a convex relaxation permits *certification* of global optimality; we prove that the relaxation is guaranteed to be tight when measurement noise is bounded.

model that relates the rotation axes of $\mathbf{A}$ and $\mathbf{B}$. This model enables a closed form solution to the rotation matrix estimation problem. By treating $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{X}$ as elements of $\mathrm{SE}(3)$, Park and Martin [1994] are able to formulate the rotation estimation problem as a least-squares problem with a closed-form solution. While these two-stage closed-form solvers are fast, these algorithms are sensitive to measurement noise because of the decoupling between the rotation and translation error terms.

To make HE calibration more robust to measurement noise, some methods jointly minimize the rotation and translation errors. Daniilidis [1999] uses a dual quaternion formulation and singular value decomposition (SVD) to solve the HE calibration problem. Andreff et al. [2001] relax the nonlinear HE calibration problem to a linear problem. To ensure the estimated rotation matrix is an element of $\mathrm{SO}(3)$, the estimated solution is projected onto the manifold of $\mathrm{SO}(3)$. Further, Andreff et al. [2001] present a solution to the monocular HE calibration problem. Čolaković-Bencerić et al. [2023] explore solving the joint HE calibration problem using the on-manifold optimization approach shown in Section 3.1.4. Similar to Daniilidis [1999], Wei et al. [2018] formulate the monocular HE calibration problem using dual quaternions, but their method leverages a second-order cone program to solve the problem. Heller et al. [2014] casts the cost function and constraints of the HE calibration problem as polynomials consisting of even-powered monomials. Using the Laserre hierarchy, Heller et al. [2014] perform a convex relaxation on the problem, which can be solved using any SDP solver. Giamou

Table 7.1: Summary of calibration methods for HE algorithms.

Method	Joint	Probabilistic	Certifiable	Scale Estimation
Tsai and Lenz [1989]	✗	✗	✗	✗
Park and Martin [1994]	✗	✗	✗	✗
Daniilidis [1999]	✓	✗	✗	✗
Andreff et al. [2001]	✓	✗	✗	✓
Čolaković-Bencerić et al. [2023]	✓	✗	✗	✗
Wei et al. [2018]	✓	✗	✓	✗
Heller et al. [2014]	✓	✗	✓	✗
Giamou et al. [2019]	✓	✗	✓	✗
Wodtko et al. [2021]	✓	✗	✓	✓
Nguyen and Pham [2018]	✗	✓	✗	✗
Brookshire and Teller [2011, 2012]	✓	✓	✗	✓
Ours	✓	✓	✓	✓

et al. [2019] formulate the HE calibration problem as a QCQP and solve the problem using an SDP relaxation. Wodtko et al. [2021] leverage dual quaternions to formulate the monocular HE calibration problem as a QCQP. Their algorithm solves the Lagrangian dual of this QCQP. Horaud and Dornaika [1995] compare a variety of joint and separate nonlinear optimization approaches to the HE problem. The authors conclude that joint methods provide more accurate solutions in the presence of noise than the two-stage closed-form methods. Finally, these joint methods minimize a purely geometric cost function and are not probabilistic. Consequently, the accuracy of each method depends on the distributions of the noisy translation and rotation measurements.

Recent research extends the hand-eye formulation to generic robotic platforms (e.g., self-driving vehicles Walters et al. [2019]) and noisy measurements. Nguyen and Pham [2018] formulate a probabilistic extension of Park and Martin [1994]. Brookshire and Teller [2011, 2012] develop a probabilistic, joint dual-quaternion formulation of the HE calibration problem, which they solve using a nonlinear optimization scheme. However, both methods do not provide any guarantee that they have found the optimal solution. In this chapter, we present a probabilistic formulation of the monocular HE calibration problem that allows us to certify the optimality of our solution.

7.2 Problem Formulation

In this section, we formulate our problem by extending the known-scale case found in Giamou et al. [2019]. Initially, we review the geometric constraints of the HE calibration problem. Using these constraints, we present an MLE approach to HE calibration. We conclude this section by casting the MLE approach as a homogeneous QCQP.

7.2.1 Geometric Constraints

We denote two rigidly connected sensor reference frames as $\underrightarrow{\mathcal{F}}_a$ and $\underrightarrow{\mathcal{F}}_b$. Additionally, we introduce an arbitrary fixed inertial world frame $\underrightarrow{\mathcal{F}}_w$. Since the sensors are rigidly connected, there is some constant $\mathbf{T}_{ba} \in \mathrm{SE}(3)$ that describes the transformation between $\underrightarrow{\mathcal{F}}_b$ and $\underrightarrow{\mathcal{F}}_a$. At time t_i, the sensors measure $\mathbf{T}_{wb}(t_i)$, and $\mathbf{T}_{wa}(t_i)$, which we can relate using

$$\mathbf{T}_{wb}(t_i) = \mathbf{T}_{wa}(t_i)\mathbf{T}_{ab}. \tag{7.1}$$

With basic algebraic manipulation, we can derive the central equation in the HE problem:

$$\begin{aligned}
\mathbf{T}_{wb}(t_{i+1}) &= \mathbf{T}_{wa}(t_{i+1})\mathbf{T}_{ab}, \\
\mathbf{T}_{wb}(t_i)\mathbf{T}_{wb}^{-1}(t_i)\mathbf{T}_{wb}(t_{i+1}) &= \mathbf{T}_{wa}(t_i)\mathbf{T}_{wa}^{-1}(t_i)\mathbf{T}_{wa}(t_{i+1})\mathbf{T}_{ab}, \\
\mathbf{T}_{wb}(t_i)\mathbf{B}_i &= \mathbf{T}_{wa}(t_i)\mathbf{A}_i\mathbf{T}_{ab}, \\
\mathbf{T}_{wa}^{-1}(t_i)\mathbf{T}_{wb}(t_i)\mathbf{B}_i &= \mathbf{A}_i\mathbf{T}_{ab}, \\
\mathbf{A}_i\mathbf{T}_{ab} &= \mathbf{T}_{ab}\mathbf{B}_i,
\end{aligned} \tag{7.2}$$

where $\mathbf{B}_i = \mathbf{T}_{wb}^{-1}(t_i)\mathbf{T}_{wb}(t_{i+1})$ and $\mathbf{A}_i = \mathbf{T}_{wa}^{-1}(t_i)\mathbf{T}_{wa}(t_{i+1})$. Going forward, we label $\mathbf{T}_{ab}$ as $\mathbf{X}$ for brevity. Consequently, we arrive at a well known formulation of the HE problem,

$$\mathbf{A}_i\mathbf{X} = \mathbf{X}\mathbf{B}_i. \tag{7.3}$$

The constraint shown in Equation (7.3) can be expanded and split into rotation and translation constraints.

$$\begin{bmatrix} \mathbf{R}_{\mathbf{A}_i} & \mathbf{t}_{\mathbf{A}_i} \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_{\mathbf{X}} & \mathbf{t}_{\mathbf{X}} \\ \mathbf{0} & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R}_{\mathbf{X}} & \mathbf{t}_{\mathbf{X}} \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_{\mathbf{B}_i} & \mathbf{t}_{\mathbf{B}_i} \\ \mathbf{0} & 1 \end{bmatrix}, \tag{7.4}$$

$$\mathbf{R}_{\mathbf{A}_i}\mathbf{R}_{\mathbf{X}} = \mathbf{R}_{\mathbf{X}}\mathbf{R}_{\mathbf{B}_i}, \tag{7.5}$$

$$\mathbf{R}_{\mathbf{A}_i}\mathbf{t}_{\mathbf{X}} + \mathbf{t}_{\mathbf{A}_i} = \mathbf{R}_{\mathbf{X}}\mathbf{t}_{\mathbf{B}_i} + \mathbf{t}_{\mathbf{X}}. \tag{7.6}$$

Equations (7.5) and (7.6) are the geometric translation and rotation constraints for the HE calibration problem if both sensors are pose sensors.

For our problem, one sensor is a scaled pose sensor, so we need to incorporate the unknown scale factor into the HE geometric constraints. The four possible translation constraints with the unknown scale factor are

$$\mathbf{R}_{\mathbf{A}_i}\mathbf{t}_{\mathbf{X}} + \gamma\mathbf{t}_{\mathbf{A}_i} = \mathbf{R}_{\mathbf{X}}\mathbf{t}_{\mathbf{B}_i} + \mathbf{t}_{\mathbf{X}}, \tag{7.7}$$

$$\frac{1}{\gamma}\mathbf{R}_{\mathbf{A}_i}\mathbf{t}_{\mathbf{X}} + \mathbf{t}_{\mathbf{A}_i} = \frac{1}{\gamma}\mathbf{R}_{\mathbf{X}}\mathbf{t}_{\mathbf{B}_i} + \frac{1}{\gamma}\mathbf{t}_{\mathbf{X}}, \tag{7.8}$$

$$\mathbf{R}_{\mathbf{A}_i}\mathbf{t}_{\mathbf{X}} + \mathbf{t}_{\mathbf{A}_i} = \gamma\mathbf{R}_{\mathbf{X}}\mathbf{t}_{\mathbf{B}_i} + \mathbf{t}_{\mathbf{X}}, \tag{7.9}$$

$$\frac{1}{\gamma}\mathbf{R}_{\mathbf{A}_i}\mathbf{t}_{\mathbf{X}} + \frac{1}{\gamma}\mathbf{t}_{\mathbf{A}_i} = \mathbf{R}_{\mathbf{X}}\mathbf{t}_{\mathbf{B}_i} + \frac{1}{\gamma}\mathbf{t}_{\mathbf{X}}, \tag{7.10}$$

where γ is the factor that 'corrects' the scaled translation. Equations (7.7) and (7.8) assume that sensor a is the scaled pose sensor, while Equations (7.9) and (7.10) assume that sensor b is the scaled pose sensor. Further, Equations (7.7) and (7.9) correct the scaled translation, while Equations (7.8) and (7.10) scale the system and pose sensor translation. To ensure that the geometric constraints are linear with respect to the unknown variables, we combine the unknown variables. As a result, Equations (7.7) to (7.10) become

$$\mathbf{R}_{\mathbf{A}_i}\mathbf{t}_{\mathbf{X}} + \gamma\mathbf{t}_{\mathbf{A}_i} = \mathbf{R}_{\mathbf{X}}\mathbf{t}_{\mathbf{B}_i} + \mathbf{t}_{\mathbf{X}}, \tag{7.11}$$

$$\mathbf{R}_{\mathbf{A}_i}\mathbf{t}_{\mathbf{X},\alpha} + \mathbf{t}_{\mathbf{A}_i} = \mathbf{R}_{\mathbf{X},\alpha}\mathbf{t}_{\mathbf{B}_i} + \mathbf{t}_{\mathbf{X},\alpha}, \tag{7.12}$$

$$\mathbf{R}_{\mathbf{A}_i}\mathbf{t}_{\mathbf{X}} + \mathbf{t}_{\mathbf{A}_i} = \mathbf{R}_{\mathbf{X},\gamma}\mathbf{t}_{\mathbf{B}_i} + \mathbf{t}_{\mathbf{X}}, \tag{7.13}$$

$$\mathbf{R}_{\mathbf{A}_i}\mathbf{t}_{\mathbf{X},\alpha} + \alpha\mathbf{t}_{\mathbf{A}_i} = \mathbf{R}_{\mathbf{X}}\mathbf{t}_{\mathbf{B}_i} + \mathbf{t}_{\mathbf{X},\alpha}, \tag{7.14}$$

where $\alpha = \frac{1}{\gamma}$ and scaled variables have the subscripts α or γ. In Equations (7.12) and (7.13), the translation vector, $\mathbf{t}_{\mathbf{B}_i}$, is a measurement, so we must combine, or scale, the unknown rotation, $\mathbf{R}_{\mathbf{X}}$, with α or γ, respectively. To align with Giamou et al. [2019], we do not use equations with scaled rotations. Unlike Equations (7.12) and (7.13), the rotation matrix, $\mathbf{R}_{\mathbf{A}_i}$, in Equation (7.14) is a measurement, so we arrive at a linear cost function by scaling the unknown translation vector, $\mathbf{t}_{\mathbf{X}}$. We choose to use either Equation (7.11) or (7.14) in our HE calibration problem.

7.2.2 Maximum Likelihood Estimation

Herein, we justify our choice of using Equation (7.14) to model the scaled translation and derive our optimization problem starting from an MLE problem instance. To start, we assume that the pose sensor measurements are noiseless. Using Equations (7.11) and (7.14), the scaled translation measurement models are, respectively,

$$\mathbf{t}_{\mathbf{A}_i} = \frac{1}{\gamma}\left(\mathbf{R}_{\mathbf{X}}\mathbf{t}_{\mathbf{B}_i} + \mathbf{t}_{\mathbf{X}} - \mathbf{R}_{\mathbf{A}_i}\mathbf{t}_{\mathbf{X}}\right) + \mathbf{n}_t(t_i), \tag{7.15}$$

$$\mathbf{t}_{\mathbf{B}_i} = \mathbf{R}_{\mathbf{X}}{}^{\intercal}\left(\mathbf{R}_{\mathbf{A}_i}\mathbf{t}_{\mathbf{X},\alpha} + \alpha\mathbf{t}_{\mathbf{A}_i} - \mathbf{t}_{\mathbf{X},\alpha}\right) + \mathbf{n}_t(t_i), \tag{7.16}$$

where $\mathbf{n}_t(t_i) \sim \mathcal{N}\left(0, {\sigma_t}^2\mathbf{I}_3\right)$ and σ_t is the standard deviation of the scaled translation measurement. To ensure that the error equations are linear with respect to our unknown variables, we must multiply Equation (7.15) by γ and Equation (7.16) by $\mathbf{R}_{\mathbf{X}}$. Multiplying Equation (7.15) by γ scales $\mathbf{n}_c$ by γ, which we cannot account for in a QCQP. However, we can multiply Equation (7.16) by $\mathbf{R}_{\mathbf{X}}$ without modifying the subsequent error distribution. As a result, we choose to use Equation (7.16) for our optimization problem. Finally, the measurement model for the rotation component of the scaled pose measurement is

$$\mathbf{R}_{\mathbf{B}_i} = \mathbf{R}_{\mathbf{Y}}{}^{\intercal}\mathbf{R}_{\mathbf{A}}\mathbf{R}_{\mathbf{X}}\mathbf{R}_n(t_i), \tag{7.17}$$

where $\mathbf{R}_n(t_i) \sim \text{Lang}(\mathbf{I}, \kappa)$. Notably, this rotation measurement model includes a right perturbation that is drawn from a Langevin distribution. In this subsection, we show that this choice enables a quadratic cost function, which we cannot achieve with the left Lie algebra perturbation approach in Chapter 5.

In this MLE problem, we wish to find

$$\{\mathbf{X}^\star, \alpha^\star\} = \underset{\mathbf{X}\in\text{SE}(3),\alpha\in\mathbb{R}}{\text{argmax}}\; p(\{\mathbf{B}_1, \ldots, \mathbf{B}_N\}|\mathbf{X}, \mathbf{A}_i). \tag{7.18}$$

We assume that all noise variables $\mathbf{R}_n(t_i)$ and $\mathbf{n}_t(t_i)$ are uncorrelated for $i = 1, \ldots, N$. The problem becomes

$$\{\mathbf{X}^\star, \alpha^\star\} = \underset{\mathbf{X}\in\text{SE}(3),\alpha\in\mathbb{R}}{\text{argmin}} -\sum_{i=1}^{N}(\log(p(\mathbf{t}_{\mathbf{B}_i}|\mathbf{X}, \mathbf{A}_i)) + \log(p(\mathbf{R}_{\mathbf{B}_i}|\mathbf{X}, \mathbf{A}_i))), \tag{7.19}$$

where we have also computed the negative log likelihood of the MLE problem. The log likelihood of $\mathbf{t}_{\mathbf{B}_i}$ is

$$\log(p(\mathbf{t}_{\mathbf{B}_i}|\mathbf{X}, \mathbf{A}_i)) = -\frac{1}{2{\sigma_c}^2}\left\|\mathbf{R}_{\mathbf{X}}{}^{\intercal}\left(\mathbf{R}_{\mathbf{A}_i}\mathbf{t}_{\mathbf{X},\alpha} + \alpha\mathbf{t}_{\mathbf{A}_i} - \mathbf{t}_{\mathbf{X},\alpha}\right) - \mathbf{t}_{\mathbf{B}_i}\right\|_2^2. \tag{7.20}$$

With some algebraic manipulation, the log-likelihood becomes

$$\log(p(\mathbf{t}_{\mathbf{B}_i}|\mathbf{X}, \mathbf{A}_i)) = -\frac{1}{2{\sigma_c}^2}\left\|\mathbf{R}_{\mathbf{A}_i}\mathbf{t}_{\mathbf{X},\alpha} + \alpha\mathbf{t}_{\mathbf{A}_i} - \mathbf{R}_{\mathbf{X}}\mathbf{t}_{\mathbf{B}_i} - \mathbf{t}_{\mathbf{X},\alpha}\right\|_2^2. \tag{7.21}$$

Following the derivation in Rosen et al. [2019], the log likelihood of $p(\mathbf{R}_{\mathbf{B}_i}|\mathbf{X}, \mathbf{A}_i)$ is

$$\log p(\mathbf{R}_{\mathbf{B}_i}|\mathbf{X}, \mathbf{A}_i) = -c(\kappa) + \kappa\text{tr}\left(\mathbf{R}_{\mathbf{X}}{}^{\intercal}\mathbf{R}_{\mathbf{A}_i}{}^{\intercal}\mathbf{R}_{\mathbf{X}}\mathbf{R}_{\mathbf{B}_i}\right), \tag{7.22}$$

$$\log p(\mathbf{R}_{\mathbf{B}_i}|\mathbf{X}, \mathbf{A}_i) = -c(\kappa) - \frac{\kappa}{2}\left\|\mathbf{R}_{\mathbf{A}_i}\mathbf{R}_{\mathbf{X}} - \mathbf{R}_{\mathbf{X}}\mathbf{R}_{\mathbf{B}_i}\right\|_{\text{F}}^2 + 3, \tag{7.23}$$

where $\text{tr}\left(\mathbf{A}^{\intercal}\mathbf{B}\right) = d - \frac{1}{2}\left\|\mathbf{A} - \mathbf{B}\right\|_F^2$. Finally, we arrive at our MLE problem.

Problem 12. *Monocular Hand-Eye Calibration as a QCQP.*

$$\begin{aligned} &\min_{\mathbf{R}_{\mathbf{X}},\mathbf{t}_{\mathbf{X},\alpha},\alpha} J_{\mathbf{t}} + J_{\mathbf{R}}, \\ &\text{s.t. } \mathbf{R}_{\mathbf{X}} \in \text{SO}(3), \end{aligned} \tag{7.24}$$

where

$$J_{\mathbf{R}} = \sum_{i=1}^{N} \kappa \left\| \mathbf{R}_{\mathbf{A}_i} \mathbf{R}_{\mathbf{X}} - \mathbf{R}_{\mathbf{X}} \mathbf{R}_{\mathbf{B}_i} \right\|_{\mathrm{F}}^2,$$
$$J_{\mathbf{t}} = \sum_{i=1}^{N} \frac{1}{{\sigma_t}^2} \left\| \mathbf{R}_{\mathbf{A}_i} \mathbf{t}_{\mathbf{X},\alpha} + \alpha \mathbf{t}_{\mathbf{A}_i} - \mathbf{R}_{\mathbf{X}} \mathbf{t}_{\mathbf{B}_i} - \mathbf{t}_{\mathbf{X},\alpha} \right\|_2^2.$$

7.2.3 Standard QCQP Formulation

Notably, Problem 12 is not in the standard QCQP form. With use of the identity $\mathbf{AXB} = (\mathbf{B}^{\mathsf{T}} \otimes \mathbf{A}) \operatorname{vec}(\mathbf{X})$ from Fackler [2005], the cost function of Problem 12 becomes

$$\begin{aligned}
\mathbf{M}_{\mathbf{R}}^{i} &= \begin{bmatrix} \mathbf{0}_{9\times 4} & \mathbf{I}_3 \otimes \mathbf{R}_{\mathbf{A}_i} - \mathbf{R}_{\mathbf{B}_i}{}^{\mathsf{T}} \otimes \mathbf{I}_3 \end{bmatrix}, \\
J_{\mathbf{R}} &= \kappa \sum_{i=1}^{N} \mathbf{x}^{\mathsf{T}} \mathbf{M}_{\mathbf{R}}^{i}{}^{\mathsf{T}} \mathbf{M}_{\mathbf{R}}^{i} \mathbf{x}, \\
\mathbf{M}_{\mathbf{t}}^{i} &= \begin{bmatrix} \mathbf{R}_{\mathbf{A}_i} - \mathbf{I}_3 & \mathbf{t}_{\mathbf{A}_i} & -\mathbf{t}_{\mathbf{B}_i}{}^{\mathsf{T}} \otimes \mathbf{I}_3 \end{bmatrix}, \\
J_{\mathbf{t}} &= \frac{1}{{\sigma_t}^2} \sum_{i=1}^{N} \mathbf{x}^{\mathsf{T}} \mathbf{M}_{\mathbf{t}}^{i\mathsf{T}} \mathbf{M}_{\mathbf{t}}^{i} \mathbf{x}, \\
\mathbf{x}^{\mathsf{T}} &= \begin{bmatrix} \mathbf{t}_{\mathbf{X},\alpha}{}^{\mathsf{T}} & \alpha & \mathbf{r}_{\mathbf{X}}{}^{\mathsf{T}} \end{bmatrix}, \\
J_{\mathbf{R}} + J_{\mathbf{t}} &= \mathbf{x}^{\mathsf{T}} \mathbf{Q} \mathbf{x},
\end{aligned} \tag{7.25}$$

where $\mathbf{r}_{\mathbf{X}} = \operatorname{vec}(\mathbf{R}_{\mathbf{X}})$ and the symmetric cost matrix $\mathbf{Q}$ can be subdivided into

$$\mathbf{Q} = \begin{bmatrix} \mathbf{Q}_{(1:4,1:4)} & \mathbf{Q}_{(1:4,5:13)} \\ \mathbf{Q}_{(1:4,5:13)}{}^{\mathsf{T}} & \mathbf{Q}_{(5:13,5:13)} \end{bmatrix}. \tag{7.26}$$

For brevity, we label $\mathbf{Q}_{(1:4,1:4)} = \mathbf{Q}_1$, $\mathbf{Q}_{(1:4,5:13)} = \mathbf{Q}_2$, and $\mathbf{Q}_{(5:13,5:13)} = \mathbf{Q}_3$. Given an optimal rotation matrix $\mathbf{R}_{\mathbf{X}}^{\star}$, the unconstrained optimal translation vector $\mathbf{t}_{\mathbf{X},\alpha}^{\star}$ and scale $\alpha^{\star}$ can be recovered by solving the linear system induced by Equation (7.26),

$$\begin{bmatrix} \mathbf{t}_{\mathbf{X},\alpha}^{\star} & \alpha^{\star} \end{bmatrix} = -\mathbf{Q}_1{}^{-1} \mathbf{Q}_2 \mathbf{r}_{\mathbf{X}}^{\star}. \tag{7.27}$$

Similar to Briales and Gonzalez-Jimenez [2017b], we use the Schur complement to reduce the cost matrix to

$$\mathbf{Q}_{red} = \mathbf{Q}_3 - \mathbf{Q}_2{}^{\mathsf{T}} \mathbf{Q}_1{}^{-1} \mathbf{Q}_2. \tag{7.28}$$

Consequently, we can reduce the dimensionality of Problem 12.

Problem 13. *Reduced QCQP Formulation of Monocular Hand-Eye Calibration.*

$$\begin{aligned}
\min_{\mathbf{r}_{\mathbf{X}} = \operatorname{vec}(\mathbf{R}_{\mathbf{X}})} \quad & \mathbf{r}_{\mathbf{X}}{}^{\mathsf{T}} \mathbf{Q}_{red} \mathbf{r}_{\mathbf{X}}, \\
\text{s.t.} \quad & \mathbf{R}_{\mathbf{X}} \in \mathrm{SO}(3).
\end{aligned} \tag{7.29}$$

7.2.4 Homogenization

To simplify the convex relaxation of Problem 13, the rotation matrix constraints from Equations (2.7) and (2.10) can be homogenized with scalar variable s:

$$\begin{aligned} \mathbf{R}^\mathsf{T}\mathbf{R} &= s^2\mathbf{I}_3, \\ \mathbf{R}\mathbf{R}^\mathsf{T} &= s^2\mathbf{I}_3, \\ \mathbf{R}^{(i)} \times \mathbf{R}^{(j)} &= s\mathbf{R}^{(k)}, \ i,j,k \in \text{cyclic}(1,2,3), \\ s^2 &= 1 \end{aligned} \tag{7.30}$$

This forms a set of 22 homogeneous quadratic equality constraints (i.e., six for each orthogonal constraint, three from each cyclic cross product, and one from the homogenizing variable s). Additionally, we augment the state vector to include s, such that

$$\mathbf{r}_s^\mathsf{T} = \begin{bmatrix} \text{vec}\left(\mathbf{R}_\mathbf{X}\right)^\mathsf{T} & s \end{bmatrix}, \tag{7.31}$$

and pad $\mathbf{Q}_{red}$ with zeros, such that

$$\mathbf{Q}_s = \begin{bmatrix} \mathbf{Q}_{red} & \mathbf{0} \\ \mathbf{0}^\mathsf{T} & 0 \end{bmatrix}. \tag{7.32}$$

7.3 Solving the Non-Convex QCQP

Following the standard procedure outlined in Boyd and Vandenberghe [2004], we derive the Lagrangian dual relaxation of the problem (see Giamou et al. [2019] and Briales and Gonzalez-Jimenez [2017b] for detailed treatments of QCQPs similar to ours). Using the homogenized $\text{SO}(3)$ constraints of Equation (7.30) from Section 7.2.4, the Lagrangian function $L(\mathbf{r}_s, \boldsymbol{\nu})$ of Problem 13 has the form

$$\begin{aligned} L(\mathbf{r}_s, \boldsymbol{\nu}) &= \nu_s + \mathbf{r}_s^\mathsf{T}\mathbf{Z}(\boldsymbol{\nu})\mathbf{r}_s, \\ \mathbf{Z}(\boldsymbol{\nu}) &= \mathbf{Q}_s + \mathbf{P}_1(\boldsymbol{\nu}) + \mathbf{P}_2(\boldsymbol{\nu}), \end{aligned} \tag{7.33}$$

where

$$\begin{aligned} \mathbf{P}_1(\boldsymbol{\nu}) &= \begin{bmatrix} -\boldsymbol{\mathcal{V}}_1 \otimes \mathbf{I}_3 - \mathbf{I}_3 \otimes \boldsymbol{\mathcal{V}}_2 & \mathbf{0}_{9\times 1} \\ \mathbf{0}_{1\times 9} & \text{tr}\left(\boldsymbol{\mathcal{V}}_1\right) + \text{tr}\left(\boldsymbol{\mathcal{V}}_2\right) \end{bmatrix}, \\ \mathbf{P}_2(\boldsymbol{\nu}) &= \begin{bmatrix} \mathbf{0}_{3\times 3} & -\boldsymbol{\nu}_{ijk}^\wedge & \boldsymbol{\nu}_{kij}^\wedge & -\boldsymbol{\nu}_{jki} \\ \boldsymbol{\nu}_{ijk}^\wedge & \mathbf{0}_{3\times 3} & -\boldsymbol{\nu}_{jki}^\wedge & -\boldsymbol{\nu}_{kij} \\ -\boldsymbol{\nu}_{kij}^\wedge & \boldsymbol{\nu}_{jki}^\wedge & \mathbf{0}_{3\times 3} & -\boldsymbol{\nu}_{ijk} \\ -\boldsymbol{\nu}_{jki}^\mathsf{T} & -\boldsymbol{\nu}_{kij}^\mathsf{T} & -\boldsymbol{\nu}_{ijk}^\mathsf{T} & -\nu_s \end{bmatrix}, \\ \boldsymbol{\mathcal{V}}_1,\, \boldsymbol{\mathcal{V}}_2 &\in \mathbb{S}^3,\ \boldsymbol{\nu}_{ijk},\, \boldsymbol{\nu}_{jki},\, \boldsymbol{\nu}_{kij} \in \mathbb{R}_3, \end{aligned} \tag{7.34}$$

and where $\mathbb{S}^3$ is the set of all 3×3 real symmetric matrices and $\boldsymbol{\nu} \in \mathbb{R}^{22}$ is a vector containing all dual variables. Next, we minimize the Lagrangian function with respect to $\mathbf{r}_s$:

$$\min_{\mathbf{r}_s} \ L(\mathbf{r}_s, \boldsymbol{\nu}) = \begin{cases} \nu_s & \mathbf{Z}(\boldsymbol{\nu}) \succeq 0, \\ -\infty & \text{otherwise.} \end{cases} \tag{7.35}$$

Finally, the Lagrangian dual problem is the following SDP.

Problem 14 (Dual of Problem 13)**.**

$$\begin{aligned} &\max_{\boldsymbol{\nu}} \nu_s, \\ &\text{s.t. } \mathbf{Z}(\boldsymbol{\nu}) \succeq 0, \end{aligned} \tag{7.36}$$

where $\mathbf{Z}(\boldsymbol{\nu})$ *is defined in Equations* (7.33) *and* (7.34)*.*

Problem 14 can be efficiently solved with any generic interior-point solver for SDPs such as Sturm [1999], Andersen and Andersen [2000], and Toh et al. [1999]. Once we have found the optimal dual parameters $\boldsymbol{\nu}^\star$, the primal solution can be found by examining the Lagrangian dual (i.e., Equation (7.33)). As shown in Briales and Gonzalez-Jimenez [2017b], the $\mathbf{r}_s^\star$ that minimizes Equation (7.33) lies in the nullspace of $\mathbf{Z}(\boldsymbol{\nu})$ because $\mathbf{Z}(\boldsymbol{\nu})$ is positive semidefinite (PSD). Since we enforce that $s = 1$, the optimal rotation is actually $\mathbf{r}_s^\star / s^\star$. We can recover $\mathbf{R}_\mathbf{X}^\star$ by wrapping, in column-wise order, the first nine entries of $\mathbf{r}_s^\star / s^\star$ into a matrix with three rows and columns. The estimated $\mathbf{t}_{\mathbf{X},\alpha}^\star$ and $\alpha^\star$ can be recovered with Equation (7.27). Finally, the actual estimated translation $\mathbf{t}_\mathbf{X}^\star$ can be recovered with $\mathbf{t}_{\mathbf{X},\alpha}^\star / \alpha^\star$. Crucially, our approach is *certifiable* because a duality gap (i.e., the difference between the primal cost and the dual cost) of zero for a candidate solution pair $\boldsymbol{\nu}^\star, \mathbf{r}^\star$ is a post-hoc guarantee or certificate of its global optimality.

7.4 Simulation Studies

To empirically demonstrate the strong duality guarantees of our problem, we focus primarily on synthetic experiments where measurement statistics and the ground truth value of $\mathbf{T}_{ab}$ are known exactly. We also compare our convex relaxation approach against a simple method that does not guarantee a global minimum on the same synthetic data. Throughout this section, 'optimality' refers specifically to global optimality of an extrinsic transformation estimate with respect to the cost function of Problem 12. In the presence of noise, optimality does not imply zero error with respect to the ground truth $\mathbf{T}_{ab}$; the dual solutions found are globally optimal with respect to Problem 12 but still differ from ground truth. Throughout all experiments, the runtime of our algorithm was on the order of two seconds without tuning optimization parameters, which is similar to the performance of its predecessor as reported in Giamou et al. [2019].

7.4.1 Optimality Certification Conditions

Throughout our experiments, three criteria are used to certify that the solution to the dual problem is optimal. First, an SVD is performed to evaluate the numerical rank of the solution matrix $\mathbf{Z}(\boldsymbol{\nu})$ in Problem 14. Any right-singular vector with corresponding singular value less than 10^{-3} is used to form the solution to the primal problem. Next, the extracted rotation solutions are checked via $\|\mathbf{R}_\mathbf{X}^\intercal \mathbf{R}_\mathbf{X} - \mathbf{I}_3\|_\mathrm{F} < 10^{-3}$, which ensures that the solution belongs to $\mathrm{SO}(3)$. Finally, solutions with a duality gap greater than 0.01% of the primal cost are rejected.[3]

7.4.2 Synthetic Data

The simulation data were created by generating trajectories on a smooth, undulating surface. The x-axis of $\underrightarrow{\mathcal{F}}_a$ was set to be tangent to the trajectory at every point $\mathbf{t}_w^{aw}(t)$, while the z-axis was set to be normal to the

[3] The approximations of floating point arithmetic necessitate the use of numerical tolerances for 'zero' singular values and duality gaps; 10^{-3} and 0.01% performed well experimentally for all problem instances tested.

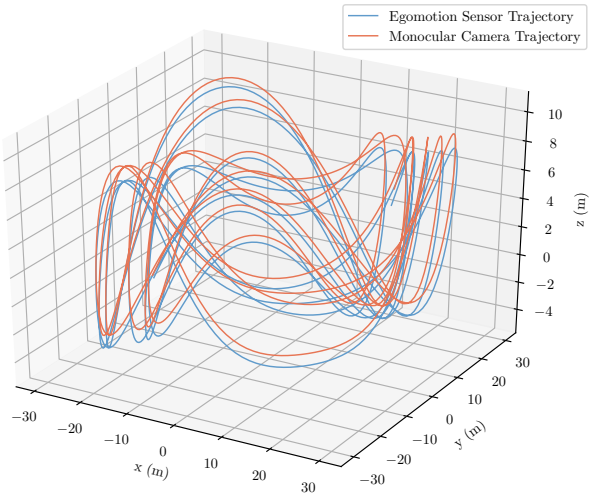


Figure 7.2: Trajectory of the rigid body system described in Section 7.4.2. Rotations of the rigid body were about all 3 axes with magnitudes ranging from 0.05 rad to 0.3 rad. The $\mathrm{SE}(3)$ trajectory of each sensor on the rigid body is corrupted by synthetic noise and the translation estimates of the monocular camera are scaled by $\alpha > 0$. The sensor platform trajectory provides sufficient rotation about different axes.

surface, thus defining $\mathbf{R}_{wa}(t)$. The full pose $\mathbf{T}_{wa}(t)$ is therefore defined by $\mathbf{R}_{wa}(t)$ and the position $\mathbf{t}_w^{aw}(t)$. The absolute position of the second sensor $\mathbf{t}_w^{bw}(t)$ at each time step was determined using the ground-truth value of the extrinsic transformation $\mathbf{T}_{ab}$ (see Equation (7.1)). An example ground-truth trajectory produced via this method is shown in Figure 7.2.

For each run, the measurements of $\mathbf{A}_i$ and $\mathbf{B}_i$ were extracted from the trajectories of sensors a and b, respectively, where b was the monocular camera. All camera translation vectors were scaled by $\alpha > 0$. Finally, we added zero-mean Gaussian noise, $\mathbf{n}_t \sim \mathcal{N}\left(\mathbf{0}, \sigma_t^2 \mathbf{I}\right)$, to each camera translation vector. Further, we injected noise into each camera rotation matrix via a left perturbation of $\mathrm{SO}(3)$. The Lie algebra of each left perturbation was drawn from $\mathbf{n}_r \sim \mathcal{N}\left(\mathbf{0}, \sigma_r^2 \mathbf{I}\right)$. The values for σ_t and σ_r are given in Table 7.2.

7.4.3 Calibration Accuracy

In this section we evaluate the performance of our algorithm against a simple, suboptimal linear approach inspired by Andreff et al. [2001]. This suboptimal approach uses the same cost function as Problem 13, but solves the unconstrained problem via SVD before projecting onto the nearest orthogonal matrix using the method of Horn et al. [1988] and then extracting the unconstrained optimal $\mathbf{t}$ and α from Equation (7.27). Since it solves successive linear least squares systems to approximately minimize the cost function, we refer to this as the linear solution. For our algorithm, we set $\kappa = 1$ and $\sigma_t = 1$. Our results are displayed in Table 7.2: each row has a different pair of translational and rotational noise settings. With minimal rotation and translation measurement noise, both algorithms perform similarly, but as the rotational measurement noise increases our solution outperforms the linear solution, highlighting the importance of a globally optimal approach. However, increases in translation noise degrade the performance of our algorithm more than the method inspired by Andreff et al. [2001].

Table 7.2: Calibration results for our monocular HE simulation study. The values in each row are estimated by a different algorithm. The mean error magnitude and standard deviation are given in each cell. The rotation standard deviations represent approximately 0.5% and 1.0% of the rotation between poses. The translation standard deviations represent approximately 0.5%, 1.0%, and 1.5% of the translation between poses.

Noise Level	Method	$t_{x,err}$ [cm]	$r_{x,err}$ [deg]	α_{err} [%]
$\sigma_r = 0.015^\circ$, $\sigma_t = 0.5$cm	Andreff	4.15 $\pm$ 2.17	0.157 $\pm$ 0.086	3.21e-4 $\pm$ 2.55e-4
	Ours	3.29 $\pm$ 1.97	0.134 $\pm$ 0.074	3.13e-4 $\pm$ 2.46e-4
$\sigma_r = 0.03^\circ$, $\sigma_t = 0.5$cm	Andreff	12.6 $\pm$ 7.4	0.833 $\pm$ 0.366	3.82e-4 $\pm$ 2.69e-4
	Ours	3.80 $\pm$ 2.05	0.152 $\pm$ 0.094	3.60e-4 $\pm$ 2.50e-4
$\sigma_r = 0.015^\circ$, $\sigma_t = 1$cm	Andreff	7.37 $\pm$ 3.60	0.162 $\pm$ 0.081	5.95e-4 $\pm$ 4.51e-4
	Ours	6.90 $\pm$ 3.69	0.227 $\pm$ 0.116	6.24e-4 $\pm$ 4.24e-4
$\sigma_r = 0.03^\circ$, $\sigma_t = 1$cm	Andreff	13.6 $\pm$ 8.14	0.823 $\pm$ 0.422	6.98e-4 $\pm$ 5.15e-4
	Ours	6.25 $\pm$ 3.29	0.24 $\pm$ 0.131	6.60e-4 $\pm$ 4.79e-4
$\sigma_r = 0.015^\circ$, $\sigma_t = 1.5$cm	Andreff	11.3 $\pm$ 6.5	0.163 $\pm$ 0.073	9.63e-4 $\pm$ 6.45e-4
	Ours	11.1 $\pm$ 6.4	0.356 $\pm$ 0.230	8.29e-4 $\pm$ 5.97e-4
$\sigma_r = 0.03^\circ$, $\sigma_t = 1.5$cm	Andreff	16.4 $\pm$ 8.3	0.818 $\pm$ 0.405	1.00e-3 $\pm$ 8.76e-4
	Ours	10.0 $\pm$ 5.7	0.377 $\pm$ 0.219	1.03e-3 $\pm$ 7.49e-4

7.5 Summary and Future Work

This chapter presented a probabilistic, certifiable monocular HE calibration algorithm with a priori global optimality guarantees. We derived a probabilistic formulation of our monocular HE calibration algorithm. Our simulation studies demonstrated that our algorithm was robust to rotation measurement noise. We leave further experimentation to future work. Additionally, we could extend our techinque using the robust problem formulation from Hu et al. [2019]. Finally, while the challenge of accurate joint spatiotemporal extrinsic calibration has been explored by Rehder et al. [2016a], Lambert et al. [2016], Marr and Kelly [2018], and Furrer et al. [2018], a certifiable algorithm has not, to our knowledge, been proposed.

Chapter 8

Certifiable Hand-Eye Robot-World Calibration

Usually, we define hand-eye robot-world (HERW) calibration in the context of a manipulator with a wrist-mounted camera that observes a stationary visual fiducial target of known scale. In this setting, HERW calibration is the process of estimating the transformations between the wrist and camera, and manipulator base and target reference frames. A common formulation of this problem is $\mathbf{AX} = \mathbf{YB}$, where our estimated transforms are $\mathbf{X}$ and $\mathbf{Y}$. Recent work from Wang et al. [2022] and Horn et al. [2023] show that this formulation can be applied to other important calibration problems that do not involve manipulators (e.g., AV camera and camera infrastructure calibration). Throughout this chapter we refer to the $\mathbf{AX} = \mathbf{YB}$ problem, as illustrated in Figure 8.1, as HERW calibration. If the manipulator has more than one wrist-mounted camera and observes multiple targets, then we can use this formulation to jointly optimize for many $\mathbf{X}$s and $\mathbf{Y}$s. While Horn et al. [2023] present a certifiable method for solving the HERW calibration problem, their method is not probabilistic and, as we show in our experiments, performs poorly on HERW calibration problems with many $\mathbf{X}$s or $\mathbf{Y}$s.

We present a probabilistic HERW calibration method that returns a certifiably optimal solution. Our method can jointly estimate many $\mathbf{X}$s and $\mathbf{Y}$s. Further, our method can solve for HERW calibration if the scale of the target is unknown, a problem that we refer to as monocular HERW calibration. This functionality enables the correction of incorrectly measured targets and unlocks the possibility of in-situ calibration. In Section 8.1, we survey HERW calibration methods. Section 8.2 formulates our HERW and monocular HERW calibration algorithms as QCQPs. We compute the Lagrangian duals of the QCQPs in Section 8.3, which we solve using a generic interior-point solver. In Section 8.4, we verify the accuracy and performance of our algorithms using a series of simulation studies and real world experiments. Finally, we summarize our findings and discuss future work in Section 8.5.

8.1 Related Work

Similar to Section 7.1, we review three categories of HERW calibration methods: separate, joint, and probabilistic. We summarize the characteristics of each method in Table 8.1.

A subset of HERW calibration methods solve for the rotation and translation components of $\mathbf{X}$ and $\mathbf{Y}$ separately. By relaxing the state set into a vector space, Li et al. [2010] convert the $\mathbf{AX} = \mathbf{YB}$ rotation cost function into a linear cost, which Li et al. [2010] minimize using singular value decomposition (SVD). After projecting

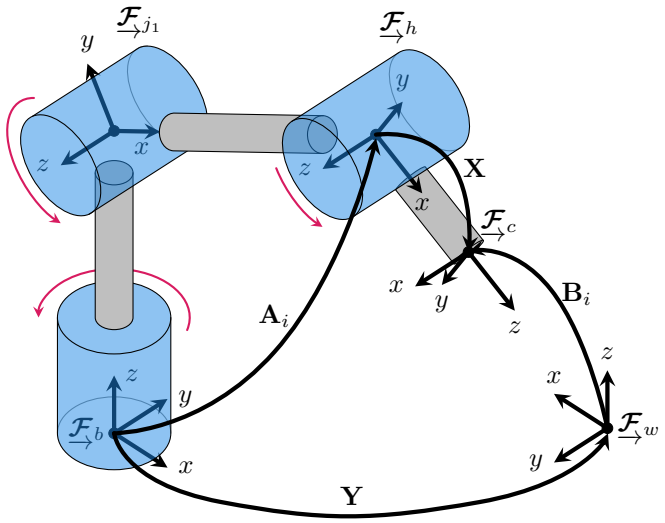


Figure 8.1: A diagram of the conventional application of HERW calibration. In this application, the objective is to estimate the transforms from the wrist of a robotic manipulator to a camera, $\mathbf{X}$, and the manipulator base to a target, $\mathbf{Y}$. In this diagram, the base, joint 1, wrist, camera, and target reference frames are labelled $\underrightarrow{\mathcal{F}}_b$, $\underrightarrow{\mathcal{F}}_{j_1}$, $\underrightarrow{\mathcal{F}}_h$, $\underrightarrow{\mathcal{F}}_c$, and $\underrightarrow{\mathcal{F}}_w$, respectively. The red arrows indicate the axis of joint rotation. At time t_i, we use the forward kinematics of the manipulator to estimate the transform from the manipulator base to the wrist, $\mathbf{A}_i$. Further, we can measure the transform from the target to the camera, $\mathbf{B}_i$.

the estimated rotation matrices onto $\mathrm{SO}(3)$, the translations of $\mathbf{X}$ and $\mathbf{Y}$ are trivial to compute. Shah [2013] examines the excitation requirements of HERW calibration. Wang et al. [2022] extend the method in Li et al. [2010] to enable estimation of multiple $\mathbf{X}$s or $\mathbf{Y}$s. Since these two-stage, closed-form solvers ignore the coupling between the rotation and translation parameters, these algorithms are sensitive to measurement noise.

To decrease the sensitivity of HERW calibration algorithms to measurement noise, some calibration methods jointly estimate the translation and rotation components of $\mathbf{X}$ and $\mathbf{Y}$. Tabb and Yousef [2017] introduce a variety of joint HERW cost functions and state parameterizations that are less sensitive to measurement noise than the two-stage closed-form HERW calibration schemes. Horn et al. [2023] cast the HERW calibration problem as a QCQP, making use of a dual quaternion parameterization, and solve a convex relaxation of the QCQP. Instead of solving the common $\mathbf{AX} = \mathbf{YB}$ formulation, Evangelista et al. [2023] develop a graph-based approach to HERW calibration using camera reprojection error terms. This formulation enables the algorithm to jointly estimate the base-target transform, end-effector-camera transform, camera intrinsic parameters, and camera distortion parameters. While each joint estimation method is more robust to measurement noise than the two-stage closed-form solvers, these methods are not probabilistic. As a result, the accuracy of the parameter estimates depends on the translation and rotation measurement noise distributions.

Table 8.1: Summary of calibration methods for HERW algorithms.

Method	Joint	Probabilistic	Certifiable	Multiple $\mathbf{X}$s and/or $\mathbf{Y}$s	Scale Estimation
Li et al. [2010]	✗	✗	✗	✗	✗
Shah [2013]	✗	✗	✗	✗	✗
Wang et al. [2022]	✗	✗	✗	$\mathbf{X}$s or $\mathbf{Y}$s	✗
Tabb and Yousef [2017]	✓	✗	✗	✗	✗
Horn et al. [2023]	✓	✗	✓	$\mathbf{X}$s or $\mathbf{Y}$s	✗
Evangelista et al. [2023]	✓	✗	✗	$\mathbf{X}$s	✗
Dornaika and Horaud [1998]	✓	✓	✗	✗	✗
Strobl and Hirzinger [2006]	✓	✓	✗	✗	✗
Ours	✓	✓	✓	$\mathbf{X}$s and $\mathbf{Y}$s	✓

Probabilistic HERW calibration algorithms are presented by Dornaika and Horaud [1998], and Strobl and Hirzinger [2006]. Dornaika and Horaud [1998] present a nonlinear, probabilistic method for HERW calibration. However, the method in Dornaika and Horaud [1998] cannot be considered an on-manifold approach because the estimated rotation matrices are constrained to $\mathrm{SO}(3)$ using penalty functions. Finally, Strobl and Hirzinger [2006] treat the HERW calibration problem as an iteratively reweighted nonlinear optimization problem, where the translation and rotation errors are corrupted by zero-mean Gaussian distributed noise. While both methods account for the probabilistic nature of the problem, neither method can provide a certificate of optimality.

8.2 Problem Formulation

In this section, we formulate our method for standard and monocular HERW calibration. Initially, we review important geometric constraints and present an MLE approach to solving both calibration problems. After converting the two problems into their standard QCQP forms, we extend the QCQPs to include many $\mathbf{X}$s and $\mathbf{Y}$s. Finally, we discuss a technique to reduce the dimensionality of HERW calibration problems with many $\mathbf{X}$s and $\mathbf{Y}$s.

8.2.1 Geometric Constraints

To start, we define the HERW equations in the context of calibrating a robot manipulator with a wrist mounted camera that observes a stationary visual fiducial target of known scale. Similar to the HE calibration problem, we assume that sensor b is the monocular camera. We justify this assumption in Section 8.2.2. We identify four reference frames $\underrightarrow{\mathcal{F}}_b, \underrightarrow{\mathcal{F}}_h, \underrightarrow{\mathcal{F}}_w, \underrightarrow{\mathcal{F}}_c$ that are associated with the base of the robot manipulator, wrist of the robot manipulator, stationary visual fiducial target reference frame, and camera reference frame, respectively. At time t_i, we use the encoder data of the arm to estimate the transform $\mathbf{T}_{bh}(t_i)$. Using the fiducial target, we estimate the camera pose relative to the target, $\mathbf{T}_{wc}(t_i)$. Our goal is to estimate the two transforms $\mathbf{T}_{hc}$ and $\mathbf{T}_{bw}$, which are related by

$$\mathbf{T}_{bh}(t_i)\mathbf{T}_{hc} = \mathbf{T}_{bw}\mathbf{T}_{wc}(t_i). \tag{8.1}$$

Generically, this geometric constraint is written as

$$\mathbf{A}_i\mathbf{X} = \mathbf{Y}\mathbf{B}_i. \tag{8.2}$$

Equation (8.2) applies to a set of four poses that can form a closed cycle (i.e., $\mathbf{X}^{-1}\mathbf{A}_i^{-1}\mathbf{Y}\mathbf{B}_i = \mathbf{I}$). In this closed cycle, the poses must alternate between being elements of the state set and measurements.

For brevity and consistency with prior literature, we continue our derivation using the $\mathbf{A}_i\mathbf{X} = \mathbf{Y}\mathbf{B}_i$ notation. We can split the rotation and translation constraints in Equation (8.2) to

$$\mathbf{R}_{\mathbf{A}_i}\mathbf{R}_{\mathbf{X}} = \mathbf{R}_{\mathbf{Y}}\mathbf{R}_{\mathbf{B}_i}, \tag{8.3}$$

$$\mathbf{R}_{\mathbf{A}_i}\mathbf{t}_{\mathbf{X}} + \mathbf{t}_{\mathbf{A}_i} = \mathbf{R}_{\mathbf{Y}}\mathbf{t}_{\mathbf{B}_i} + \mathbf{t}_{\mathbf{Y}}, \tag{8.4}$$

where $\mathbf{R}_{\mathbf{A}_i}, \mathbf{R}_{\mathbf{X}}, \mathbf{R}_{\mathbf{Y}}, \mathbf{R}_{\mathbf{B}_i}$ and $\mathbf{t}_{\mathbf{A}_i}, \mathbf{t}_{\mathbf{X}}, \mathbf{t}_{\mathbf{Y}}, \mathbf{t}_{\mathbf{B}_i}$ are the rotation and translation components of $\mathbf{A}_i, \mathbf{X}, \mathbf{Y}, \mathbf{B}_i$, respectively.

8.2.2 Maximum Likelihood Estimation

Using the split rotation and translation constraints, we can formulate an MLE problem. Similar to our derivation in Section 7.2.2, we assume that $\mathbf{R}_{\mathbf{A}_i}$ and $\mathbf{t}_{\mathbf{A}_i}$ are noise free. We model the camera translation measurement as

$$\mathbf{t}_{\mathbf{B}_i} = \mathbf{R}_{\mathbf{Y}}{}^{\mathsf{T}}(\mathbf{R}_{\mathbf{A}_i}\mathbf{t}_{\mathbf{X}} + \mathbf{t}_{\mathbf{A}_i} - \mathbf{t}_{\mathbf{Y}}) + \mathbf{n}_t(t_i), \tag{8.5}$$

where $\mathbf{n}_t(t_i) \sim \mathcal{N}\left(0, {\sigma_t}^2\mathbf{I}_3\right)$ and σ_t is the standard deviation of the translation component of measurement $\mathbf{B}_i$. Finally, we model the camera rotation measurement as

$$\mathbf{R}_{\mathbf{B}_i} = \mathbf{R}_{\mathbf{Y}}{}^{\mathsf{T}}\mathbf{R}_{\mathbf{A}_i}\mathbf{R}_{\mathbf{X}}\mathbf{R}_n(t_i), \tag{8.6}$$

where $\mathbf{R}_n(t_i) \sim \text{Lang}(\mathbf{I}_3, \kappa)$, where κ is the concentration of the rotations of the measurement $\mathbf{B}_i$.

Our optimization problem is

$$\{\mathbf{X}^\star, \mathbf{Y}^\star\} = \underset{\mathbf{X},\mathbf{Y}\in\text{SE}(3)}{\text{argmax}}\ p(\{\mathbf{B}_1, \ldots, \mathbf{B}_N\}|\mathbf{X}, \mathbf{Y}). \tag{8.7}$$

We assume that all noise variables $\mathbf{R}_n(t_i)$ and $\mathbf{n}_t(t_i)$ are uncorrelated for $i = 1, \ldots, N$. The negative log likelihood of the problem is

$$\{\mathbf{X}^\star, \mathbf{Y}^\star\} = \underset{\mathbf{X},\mathbf{Y}\in\text{SE}(3)}{\text{argmin}} -\sum_{i=1}^{N}(\log(p(\mathbf{t}_{\mathbf{B}_i}|\mathbf{X}, \mathbf{Y})) + \log(p(\mathbf{R}_{\mathbf{B}_i}|\mathbf{X}, \mathbf{Y}))). \tag{8.8}$$

The log likelihood of $\mathbf{t}_{\mathbf{B}}$ is

$$\log(p(\mathbf{t}_{\mathbf{B}_i}|\mathbf{X}, \mathbf{Y})) = -\frac{1}{2{\sigma_t}^2}\left\|\mathbf{R}_{\mathbf{Y}}{}^T(\mathbf{R}_{\mathbf{A}_i}\mathbf{t}_{\mathbf{X}} + \mathbf{t}_{\mathbf{A}_i} - \mathbf{t}_{\mathbf{Y}}) - \mathbf{t}_{\mathbf{B}_i}\right\|_2^2 \tag{8.9}$$

Since the covariance of $\log(p(\mathbf{t}_{\mathbf{B}_i}|\mathbf{X}, \mathbf{Y}))$ is isotropic, we can make the log likelihood a function of only quadratic terms, that is

$$\log(p(\mathbf{t}_{\mathbf{B}_i}|\mathbf{X}, \mathbf{Y})) = -\frac{1}{2{\sigma_t}^2}\left\|\mathbf{R}_{\mathbf{A}_i}\mathbf{t}_{\mathbf{X}} + \mathbf{t}_{\mathbf{A}_i} - \mathbf{t}_{\mathbf{Y}} - \mathbf{R}_{\mathbf{Y}}\mathbf{t}_{\mathbf{B}_i}\right\|_2^2. \tag{8.10}$$

Following the derivation in Rosen et al. [2019], the log likelihood of $p(\mathbf{R}_{\mathbf{B}_i}|\mathbf{X}, \mathbf{Y})$ is

$$\log p(\mathbf{R}_{\mathbf{B}_i}|\mathbf{X}, \mathbf{Y}) = -c(\kappa) + \kappa\text{tr}\left(\mathbf{R}_{\mathbf{X}}{}^{\mathsf{T}}\mathbf{R}_{\mathbf{A}_i}{}^{\mathsf{T}}\mathbf{R}_{\mathbf{Y}}\mathbf{R}_{\mathbf{B}_i}\right), \tag{8.11}$$

$$\log p(\mathbf{R}_{\mathbf{B}_i}|\mathbf{X}, \mathbf{Y}) = -c(\kappa) - \frac{\kappa}{2}\left\|\mathbf{R}_{\mathbf{A}_i}\mathbf{R}_{\mathbf{X}} - \mathbf{R}_{\mathbf{Y}}\mathbf{R}_{\mathbf{B}_i}\right\|_{\text{F}}^2 + 3, \tag{8.12}$$

where $\text{tr}\left(\mathbf{A}^{\mathsf{T}}\mathbf{B}\right) = d - \frac{1}{2}\left\|\mathbf{A} - \mathbf{B}\right\|_F^2$. Finally, our optimization problem is as follows.

Problem 15. *QCQP formulation of the Standard HERW Problem*

$$\min_{\mathbf{R}_{\mathbf{X}},\mathbf{R}_{\mathbf{Y}}\in\text{SO}(3),\ \mathbf{t}_{\mathbf{X}},\mathbf{t}_{\mathbf{Y}}\in\mathbb{R}^3} J_{\mathbf{t}} + J_{\mathbf{R}}, \tag{8.13}$$

$$\text{s.t. } \mathbf{R}_{\mathbf{X}}, \mathbf{R}_{\mathbf{Y}} \in \text{SO}(3). \tag{8.14}$$

where

$$J_{\mathbf{R}} = \kappa \sum_{i=0}^{N} \|\mathbf{R}_{\mathbf{A}_i}\mathbf{R}_{\mathbf{X}} - \mathbf{R}_{\mathbf{Y}}\mathbf{R}_{\mathbf{B}_i}\|_{\mathrm{F}}^2, \tag{8.15}$$

$$J_{\mathbf{t}} = \frac{1}{\sigma_t^2} \sum_{i=0}^{N} \|\mathbf{R}_{\mathbf{A}_i}\mathbf{t}_{\mathbf{X}} + \mathbf{t}_{\mathbf{A}_i} - \mathbf{R}_{\mathbf{Y}}\mathbf{t}_{\mathbf{B}_i} - \mathbf{t}_{\mathbf{Y}}\|_2^2. \tag{8.16}$$

To ensure that all of the terms are quadratic, we homogenize the Equation (8.16) using the homogenizing variable s such that

$$J_{\mathbf{t}} = \frac{1}{\sigma_t^2} \sum_{i=0}^{N} \|\mathbf{R}_{\mathbf{A}_i}\mathbf{t}_{\mathbf{X}} + s\mathbf{t}_{\mathbf{A}_i} - \mathbf{R}_{\mathbf{Y}}\mathbf{t}_{\mathbf{B}_i} - \mathbf{t}_{\mathbf{Y}_i}\|_2^2, \tag{8.17}$$

and add the constraint $s^2 = 1$.

8.2.3 Targets with Unknown Scale

In Problem 15, the combination of the camera and fiducial target allows us to treat the camera as a scaled pose sensor. Following the MLE derivation from before, we modify the camera translation model to be

$$\mathbf{t}_{\mathbf{B}_i} = \alpha\mathbf{R}_{\mathbf{Y}}^{\mathsf{T}}(\mathbf{R}_{\mathbf{A}_i}\mathbf{t}_{\mathbf{X}} + \mathbf{t}_{\mathbf{A}_i} - \mathbf{t}_{\mathbf{Y}}) + \mathbf{n}_t, \tag{8.18}$$

where $\mathbf{n}_t \sim \mathcal{N}\left(0, {\sigma_t}^2\mathbf{I}\right)$ and σ_t is the standard deviation of the unscaled translation estimate. Subsequently, the MLE problem becomes the following.

Problem 16. *QCQP formulation of the Monocular HERW Problem*

$$\min_{\mathbf{R}_{\mathbf{X}},\mathbf{R}_{\mathbf{Y}}\in\mathrm{SO}(3),\ \mathbf{t}_{\mathbf{X}},\mathbf{t}_{\mathbf{Y}}\in\mathbb{R}^3,\ \alpha\in\mathbb{R}} J_{\mathbf{t},\alpha} + J_{\mathbf{R},\alpha}, \tag{8.19}$$

$$\textit{s.t. } \mathbf{R}_{\mathbf{X}}, \mathbf{R}_{\mathbf{Y}} \in \mathrm{SO}(3). \tag{8.20}$$

where

$$J_{\mathbf{R},\alpha} = \kappa \sum_{i=0}^{N} \|\mathbf{R}_{\mathbf{A}_i}\mathbf{R}_{\mathbf{X}} - \mathbf{R}_{\mathbf{Y}}\mathbf{R}_{\mathbf{B}_i}\|_{\mathrm{F}}^2, \tag{8.21}$$

$$J_{\mathbf{t},\alpha} = \frac{1}{{\sigma_t}^2} \sum_{i=0}^{N} \|\mathbf{R}_{\mathbf{A}_i}\mathbf{t}_{\mathbf{X},\alpha} + \alpha\mathbf{t}_{\mathbf{A}_i} - \mathbf{R}_{\mathbf{Y}}\mathbf{t}_{\mathbf{B}_i} - \mathbf{t}_{\mathbf{Y},\alpha}\|_2^2. \tag{8.22}$$

To maintain the first assumption of our MLE formulation in Section 8.2.2, we make sensor b the scaled pose sensor. Similar to Chapter 7, we use α to scale the estimated system and maintain linearity in our error equations. We recover the calibration parameters using $\frac{1}{\alpha}\mathbf{t}_{\mathbf{X},\alpha}$ and $\frac{1}{\alpha}\mathbf{t}_{\mathbf{Y},\alpha}$. Finally, we do not need to include a homogenizing variable in the cost function of Problem 16 because the unscaled cost is homogeneous.

8.2.4 Standard QCQP Formulation

Herein, we convert Problems 15 and 16 into their standard QCQP forms. The state vectors for the standard and monocular HERW problems are, respectively,

$$\mathbf{x}^\intercal = \begin{bmatrix} \mathbf{t}_\mathbf{X}^\intercal & \mathbf{t}_\mathbf{Y}^\intercal & \mathbf{r}_\mathbf{X}^\intercal & \mathbf{r}_\mathbf{Y}^\intercal & s \end{bmatrix}^\intercal, \tag{8.23}$$

$$\mathbf{x}_\alpha^\intercal = \begin{bmatrix} \mathbf{t}_{\mathbf{X},\alpha}^\intercal & \mathbf{t}_{\mathbf{Y},\alpha}^\intercal & \alpha & \mathbf{r}_\mathbf{X}^\intercal & \mathbf{r}_\mathbf{Y}^\intercal & s \end{bmatrix}^\intercal, \tag{8.24}$$

where $\mathbf{r}_\mathbf{X} = \mathrm{vec}\left(\mathbf{R}_\mathbf{X}\right)$ and $\mathbf{r}_\mathbf{Y} = \mathrm{vec}\left(\mathbf{R}_\mathbf{Y}\right)$. Importantly, the rotation matrix constraints require homogenization, so we have appended the homogenization variable to the state vector for the monocular HERW problem. The standard and monocular HERW rotation cost terms are

$$J_\mathbf{R} = \kappa \sum_{i=1}^{N} \mathbf{x}^\intercal \mathbf{M}_{\mathbf{R}_i}{}^\intercal \mathbf{M}_{\mathbf{R}_i} \mathbf{x}, \tag{8.25}$$

$$J_{\mathbf{R},\alpha} = \kappa \sum_{i=1}^{N} \mathbf{x}_\alpha{}^\intercal \mathbf{M}_{\mathbf{R}_i,\alpha}{}^\intercal \mathbf{M}_{\mathbf{R}_i,\alpha} \mathbf{x}_\alpha, \tag{8.26}$$

where

$$\mathbf{M}_{\mathbf{R}_i} = \begin{bmatrix} \mathbf{0}_{9\times 6} & \mathbf{I} \otimes \mathbf{R}_{\mathbf{A}_i} & -\mathbf{R}_{\mathbf{B}_i} \otimes \mathbf{I} & 0 \end{bmatrix}, \tag{8.27}$$

$$\mathbf{M}_{\mathbf{R}_i,\alpha} = \begin{bmatrix} \mathbf{0}_{9\times 7} & \mathbf{I} \otimes \mathbf{R}_{\mathbf{A}_i} & -\mathbf{R}_{\mathbf{B}_i} \otimes \mathbf{I} & 0 \end{bmatrix}. \tag{8.28}$$

The standard and monocular HERW translation cost terms are

$$J_\mathbf{t} = \frac{1}{\sigma_t^2} \sum_{i=1}^{N} \mathbf{x}^\intercal \mathbf{M}_{\mathbf{t}_i}{}^\intercal \mathbf{M}_{\mathbf{t}_i} \mathbf{x}, \tag{8.29}$$

$$J_{\mathbf{t},\alpha} = \frac{1}{\sigma_t^2} \sum_{i=1}^{N} \mathbf{x}_\alpha{}^\intercal \mathbf{M}_{\mathbf{t}_i,\alpha}{}^\intercal \mathbf{M}_{\mathbf{t}_i,\alpha} \mathbf{x}_\alpha, \tag{8.30}$$

respectively, and

$$\mathbf{M}_{\mathbf{t}_i} = \begin{bmatrix} \mathbf{R}_{\mathbf{A}_i} & -\mathbf{I} & \mathbf{0}_{3\times 9} & -\mathbf{t}_{\mathbf{B}_i}{}^\intercal \otimes \mathbf{I} & \mathbf{t}_{\mathbf{A}_i} \end{bmatrix}, \tag{8.31}$$

$$\mathbf{M}_{\mathbf{t}_i,\alpha} = \begin{bmatrix} \mathbf{R}_{\mathbf{A}_i} & -\mathbf{I} & \mathbf{t}_{\mathbf{A}_i} & \mathbf{0}_{3\times 9} & -\mathbf{t}_{\mathbf{B}_i}{}^\intercal \otimes \mathbf{I} & \mathbf{0}_{3\times 1} \end{bmatrix}. \tag{8.32}$$

Similar to the HE case, the symmetric standard and monocular HERW cost matrices are

$$\mathbf{Q} = \kappa \sum_{i=1}^{N} \mathbf{M}_{\mathbf{R}_i}{}^\intercal \mathbf{M}_{\mathbf{R}_i} + \frac{1}{\sigma_t{}^2} \sum_{i=1}^{N} \mathbf{M}_{\mathbf{t}_i}{}^\intercal \mathbf{M}_{\mathbf{t}_i}, \tag{8.33}$$

$$\mathbf{Q}_\alpha = \kappa \sum_{i=1}^{N} \mathbf{M}_{\mathbf{R}_i,\alpha}{}^\intercal \mathbf{M}_{\mathbf{R}_i,\alpha} + \frac{1}{\sigma_t{}^2} \sum_{i=1}^{N} \mathbf{M}_{\mathbf{t}_i,\alpha}{}^\intercal \mathbf{M}_{\mathbf{t}_i,\alpha}. \tag{8.34}$$

Consequently, the QCQP formulations of the standard and monocular HERW optimization problems are as shown below.

Problem 17. *QCQP formulation of the Standard HERW Problem*

$$\min_{\mathbf{R_X},\mathbf{R_Y}\in\mathrm{SO}(3),\ \mathbf{t_X},\mathbf{t_Y}\in\mathbb{R}^3} \mathbf{x}^\intercal\mathbf{Q}\mathbf{x}, \tag{8.35}$$

$$\text{s.t. } \mathbf{R_X},\mathbf{R_Y}\in\mathrm{SO}(3). \tag{8.36}$$

Problem 18. *QCQP formulation of the Monocular HERW Problem*

$$\min_{\mathbf{R_X},\mathbf{R_Y}\in\mathrm{SO}(3),\ \mathbf{t}_{\mathbf{X},\alpha},\mathbf{t}_{\mathbf{Y},\alpha}\in\mathbb{R}^3,\ \alpha\in\mathbb{R}} \mathbf{x}_\alpha^\intercal\mathbf{Q}_\alpha\mathbf{x}_\alpha, \tag{8.37}$$

$$\text{s.t. } \mathbf{R_X},\mathbf{R_Y}\in\mathrm{SO}(3). \tag{8.38}$$

8.2.5 The HERW Problem as a Bipartite Graph

Suppose we have an HERW system consisting of M $\mathbf{X}$ and P $\mathbf{Y}$ transforms. For this system, the states of the HERW and monocular HERW calibration problems are, respectively,

$$\begin{aligned}
\mathbf{x}^\intercal &= \begin{bmatrix} \mathbf{t}_{\mathbf{X}_0}^\intercal & \cdots & \mathbf{t}_{\mathbf{X}_{M-1}}^\intercal & \mathbf{t}_{\mathbf{Y}_0}^\intercal & \cdots & \mathbf{t}_{\mathbf{Y}_{P-1}}^\intercal & \mathbf{r}_{\mathbf{X}_0}^\intercal & \cdots & \mathbf{r}_{\mathbf{X}_{M-1}}^\intercal \\ \mathbf{r}_{\mathbf{Y}_0}^\intercal & \cdots & \mathbf{r}_{\mathbf{Y}_{P-1}}^\intercal & s \end{bmatrix} \\
\mathbf{x}_\alpha^\intercal &= \begin{bmatrix} \mathbf{t}_{\mathbf{X}_0}^\intercal & \cdots & \mathbf{t}_{\mathbf{X}_{M-1}}^\intercal & \mathbf{t}_{\mathbf{Y}_0}^\intercal & \cdots & \mathbf{t}_{\mathbf{Y}_{P-1}}^\intercal & \alpha & \mathbf{r}_{\mathbf{X}_0}^\intercal & \cdots & \mathbf{r}_{\mathbf{X}_{M-1}}^\intercal \\ \mathbf{r}_{\mathbf{Y}_0}^\intercal & \cdots & \mathbf{r}_{\mathbf{Y}_{P-1}}^\intercal & s \end{bmatrix},
\end{aligned}$$

Let $\mathcal{D}$ be the set of all cost matrices for each $\mathbf{X}_j$-$\mathbf{Y}_k$ pair with measurements. For example, if $\mathbf{X}_0$ and $\mathbf{Y}_0$ have a measurements for HERW calibration, then an element of $\mathcal{D}$ is the cost matrix between $\mathbf{X}_0$ and $\mathbf{Y}_0$. Unlike HE calibration, where each $\mathbf{X}$ is unrelated, our states, in HERW calibration, are connected and can influence each other. Specifically, each $\mathbf{X}$ is connected to at least one $\mathbf{Y}$ through the cost matrix. We can view these connections as forming a graph, where the nodes are the states that we wish to estimate (see Figure 8.2). Interestingly, only $\mathbf{X}$s can connect to $\mathbf{Y}$s and vice-versa, which imparts a special structure on the graph. We refer to graphs with these restrictions as bipartite graphs. As opposed to solving each HERW problem individually, we can solve this bipartite graph, which improves estimation accuracy.

Let us start our formulation of the many $\mathbf{X}$ and $\mathbf{Y}$ problem by investigating the structure of Equations (8.33)

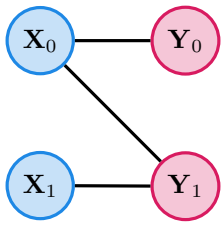


Figure 8.2: An example of a problem with multiple $\mathbf{X}$s and $\mathbf{Y}$s.

and (8.34). The standard cost matrix, $\mathbf{Q} \in \mathbb{S}^{25\times 25}$, can be divided into twelve unique components, such that

$$\mathbf{Q} = \begin{bmatrix} \mathbf{Q}_{(1:3,1:3)} & \mathbf{Q}_{(1:3,4:6)} & \mathbf{0} & \mathbf{Q}_{(1:3,16:24)} & \mathbf{Q}_{(1:3,25)} \\ \mathbf{Q}_{(1:3,4:6)}^\mathsf{T} & \mathbf{Q}_{(4:6,4:6)} & \mathbf{0} & \mathbf{Q}_{(4:6,16:24)} & \mathbf{Q}_{(4:6,25)} \\ \mathbf{0}^\mathsf{T} & \mathbf{0}^\mathsf{T} & \mathbf{Q}_{(7:15,7:15)} & \mathbf{Q}_{(7:15,16:24)} & \mathbf{0} \\ \mathbf{Q}_{(1:3,16:24)}^\mathsf{T} & \mathbf{Q}_{(4:6,16:24)}^\mathsf{T} & \mathbf{Q}_{(7:15,16:24)}^\mathsf{T} & \mathbf{Q}_{(16:24,16:24)} & \mathbf{Q}_{(16:24,25)} \\ \mathbf{Q}_{(1:3,25)}^\mathsf{T} & \mathbf{Q}_{(4:6,25)}^\mathsf{T} & \mathbf{0}^\mathsf{T} & \mathbf{Q}_{(16:24,25)}^\mathsf{T} & \mathbf{Q}_{(25,25)} \end{bmatrix}. \tag{8.39}$$

For brevity, let us label the components of the cost matrix

$$\begin{aligned} &\mathbf{Q}_0 = \mathbf{Q}_{(1:3,1:3)}, && \mathbf{Q}_1 = \mathbf{Q}_{(1:3,4:6)}, && \mathbf{Q}_2 = \mathbf{Q}_{(1:3,16:24)}, && \mathbf{Q}_3 = \mathbf{Q}_{(1:3,25)}, \\ &\mathbf{Q}_4 = \mathbf{Q}_{(4:6,4:6)}, && \mathbf{Q}_5 = \mathbf{Q}_{(4:6,16:24)}, && \mathbf{Q}_6 = \mathbf{Q}_{(4:6,25)}, && \mathbf{Q}_7 = \mathbf{Q}_{(7:15,7:15)}, \\ &\mathbf{Q}_8 = \mathbf{Q}_{(7:15,16:24)}, && \mathbf{Q}_9 = \mathbf{Q}_{(16:24,16:24)}, && \mathbf{Q}_{10} = \mathbf{Q}_{(16:24,25)}, && \mathbf{Q}_{11} = \mathbf{Q}_{(25,25)}. \end{aligned}$$

Similarly, the monocular cost matrix $\mathbf{Q}_\alpha \in \mathbb{S}^{26\times 26}$ becomes

$$\mathbf{Q}_\alpha = \begin{bmatrix} \mathbf{Q}_0 & \mathbf{Q}_1 & \mathbf{Q}_3 & \mathbf{0} & \mathbf{Q}_2 & \mathbf{0} \\ \mathbf{Q}_1^\mathsf{T} & \mathbf{Q}_4 & \mathbf{Q}_6 & \mathbf{0} & \mathbf{Q}_5 & \mathbf{0} \\ \mathbf{Q}_3^\mathsf{T} & \mathbf{Q}_6^\mathsf{T} & \mathbf{Q}_{11} & \mathbf{0} & \mathbf{Q}_{10}^\mathsf{T} & 0 \\ \mathbf{0}^\mathsf{T} & \mathbf{0}^\mathsf{T} & \mathbf{0}^\mathsf{T} & \mathbf{Q}_7 & \mathbf{Q}_8 & \mathbf{0} \\ \mathbf{Q}_2^\mathsf{T} & \mathbf{Q}_5^\mathsf{T} & \mathbf{Q}_{10} & \mathbf{Q}_8^\mathsf{T} & \mathbf{Q}_9 & \mathbf{0} \\ \mathbf{0}^\mathsf{T} & \mathbf{0}^\mathsf{T} & 0 & \mathbf{0}^\mathsf{T} & \mathbf{0}^\mathsf{T} & 0 \end{bmatrix}. \tag{8.40}$$

Notably, the standard and monocular HERW cost matrices are related by inserting the values of the last column and row to the correct locations in the seventh column and row. Going forward, we focus on the monocular HERW cost function because converting the monocular cost function into the standard cost function is trivial. We label the monocular HERW cost matrix component $\mathbf{Q}_l$ for the pair $\mathbf{X}_j$-$\mathbf{Y}_k$ as $\mathbf{Q}_{\alpha,j,k,l}$, where $l = 0, \ldots, 11$ and (j, k) is a pair with data in $\mathcal{D}$. Additionally, we define

$$b_{j,k} = \begin{cases} 1, \mathbf{Q}_{\alpha,j,k} \in \mathcal{D}, \\ 0, \mathbf{Q}_{\alpha,j,k} \notin \mathcal{D}, \end{cases} \tag{8.41}$$

where $\mathbf{Q}_{\alpha,j,k}$ is the monocular cost matrix associated with $\mathbf{X}_j$ and $\mathbf{Y}_k$. The total cost matrix is

$$\mathbf{Q}_{\alpha,tot} = \begin{bmatrix} \mathbf{U} & \mathbf{V} \\ \mathbf{V}^\mathsf{T} & \mathbf{W} \end{bmatrix}. \tag{8.42}$$

The matrix $\mathbf{U}$ is

$$\mathbf{U} = \begin{bmatrix} \mathbf{U}_0 & \mathbf{U}_1 & \mathbf{u}_0 \\ \mathbf{U}_1^\mathsf{T} & \mathbf{U}_2 & \mathbf{u}_1 \\ \mathbf{u}_0^\mathsf{T} & \mathbf{u}_1 & \sum_{j=0}^{M-1}\sum_{k=0}^{P-1} b_{j,k}\mathbf{Q}_{\alpha,j,k,11} \end{bmatrix} \tag{8.43}$$

where

$$\mathbf{U}_0 = \text{blkdiag}\left(\sum_{k=0}^{P-1} b_{0,k}\mathbf{Q}_{\alpha,0,k,0}, \ \ldots \ , \sum_{k=0}^{P-1} b_{M-1,k}\mathbf{Q}_{\alpha,M-1,k,0}\right), \tag{8.44}$$

$$\mathbf{u}_0{}^\intercal = \begin{bmatrix} \sum_{k=0}^{P-1} b_{0,k}\mathbf{Q}_{\alpha,0,k,3}{}^\intercal & \cdots & \sum_{k=0}^{P-1} b_{M-1,k}\mathbf{Q}_{\alpha,M-1,k,3}{}^\intercal \end{bmatrix}, \tag{8.45}$$

$$\mathbf{U}_1 = \begin{bmatrix} b_{0,0}\mathbf{Q}_{\alpha,0,0,1} & \cdots & b_{0,P-1}\mathbf{Q}_{\alpha,0,P-1,1} \\ \vdots & \ddots & \vdots \\ b_{M-1,0}\mathbf{Q}_{\alpha,M-1,0,1} & \cdots & b_{0,P-1}\mathbf{Q}_{\alpha,M-1,P-1,1} \end{bmatrix}, \tag{8.46}$$

$$\mathbf{u}_1{}^\intercal = \begin{bmatrix} \sum_{j=0}^{M-1} b_{j,0}\mathbf{Q}_{\alpha,j,0,6}{}^\intercal & \cdots & \sum_{j=0}^{M-1} b_{j,P-1}\mathbf{Q}_{\alpha,j,P-1,6}{}^\intercal \end{bmatrix}, \tag{8.47}$$

$$\mathbf{U}_2 = \text{blkdiag}\left(\sum_{j=0}^{M-1} b_{j,0}\mathbf{Q}_{\alpha,j,0,4}, \cdots, \sum_{j=0}^{M-1} b_{j,P-1}\mathbf{Q}_{\alpha,j,P-1,4} \right). \tag{8.48}$$

The submatrix $\mathbf{V}$ is

$$\mathbf{V} = \begin{bmatrix} \mathbf{0} & b_{0,0}\mathbf{Q}_{\alpha,0,0,2} & b_{0,1}\mathbf{Q}_{\alpha,0,1,2} & \cdots & b_{0,P-1}\mathbf{Q}_{\alpha,0,P-1,2} & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0} & b_{M-1,0}\mathbf{Q}_{\alpha,M-1,0,2} & b_{M-1,1}\mathbf{Q}_{\alpha,M-1,1,2} & \cdots & b_{M-1,P-1}\mathbf{Q}_{\alpha,M-1,P-1,2} & \mathbf{0} \\ \mathbf{0} & \sum_{j=0}^{M-1} b_{j,0}\mathbf{Q}_{\alpha,j,0,5} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \ddots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \sum_{j=0}^{M-1} b_{j,P-1}\mathbf{Q}_{\alpha,j,P-1,5} & \mathbf{0} \\ \mathbf{0}^\intercal & \sum_{j=0}^{M-1} b_{j,0}\mathbf{Q}_{\alpha,j,0,10}{}^\intercal & \sum_{j=0}^{M-1} b_{j,1}\mathbf{Q}_{\alpha,j,1,10}{}^\intercal & \cdots & \sum_{j=0}^{M-1} b_{j,P-1}\mathbf{Q}_{\alpha,j,P-1,10}{}^\intercal & 0 \end{bmatrix}. \tag{8.49}$$

Finally, the submatrix $\mathbf{W}$ is

$$\mathbf{W} = \begin{bmatrix} \mathbf{W}_0 & \mathbf{W}_1 & \mathbf{0} \\ \mathbf{W}_1{}^\intercal & \mathbf{W}_2 & \mathbf{0} \\ \mathbf{0}^\intercal & \mathbf{0}^\intercal & 0 \end{bmatrix}, \tag{8.50}$$

where

$$\mathbf{W}_0 = \text{blkdiag}\left(\sum_{k=0}^{P-1} b_{0,k}\mathbf{Q}_{\alpha,0,k,7}, \cdots, \sum_{k=0}^{P-1} b_{M-1,k}\mathbf{Q}_{\alpha,M-1,k,7} \right), \tag{8.51}$$

$$\mathbf{W}_1 = \begin{bmatrix} b_{0,0}\mathbf{Q}_{\alpha,0,0,8} & \cdots & b_{0,P-1}\mathbf{Q}_{\alpha,0,P-1,8} \\ \vdots & \ddots & \vdots \\ b_{M-1,0}\mathbf{Q}_{\alpha,M-1,0,8} & \cdots & b_{0,P-1}\mathbf{Q}_{\alpha,M-1,P-1,8} \end{bmatrix}, \tag{8.52}$$

$$\mathbf{W}_2 = \text{blkdiag}\left(\sum_{j=0}^{M-1} b_{j,0}\mathbf{Q}_{\alpha,j,0,9}, \cdots, \sum_{j=0}^{M-1} b_{j,P-1}\mathbf{Q}_{\alpha,j,P-1,9} \right). \tag{8.53}$$

As a result, the standard and monocular HERW optimization problems for many $\mathbf{X}$s and $\mathbf{Y}$s are as follows.

Problem 19. *QCQP Formulation of the Standard HERW Problem with Many $\mathbf{X}$s, and $\mathbf{Y}$s.*

$$\min_{\mathbf{x}} \mathbf{x}^\intercal \mathbf{Q}_{tot} \mathbf{x}, \tag{8.54}$$

$$\text{s.t. } \mathbf{R}_{\mathbf{X}_j} \in \text{SO}(3) \; \forall \; j = 0, \ldots, M-1, \tag{8.55}$$

$$\mathbf{R}_{\mathbf{Y}_k} \in \mathrm{SO}(3) \; \forall \; k = 0, \ldots, P-1. \tag{8.56}$$

Problem 20. *QCQP Formulation of the Monocular HERW Problem with Unknown Scale, Many* $\mathbf{X}$*s, and* $\mathbf{Y}$*s.*

$$\begin{aligned} \min_{\mathbf{x}_\alpha} \; & \mathbf{x}_\alpha{}^\mathsf{T} \mathbf{Q}_{\alpha,tot} \mathbf{x}_\alpha, && (8.57)\\ \text{s.t. } & \mathbf{R}_{\mathbf{X}_j} \in \mathrm{SO}(3) \; \forall \; j = 0, \ldots, M-1, && (8.58)\\ & \mathbf{R}_{\mathbf{Y}_k} \in \mathrm{SO}(3) \; \forall \; k = 0, \ldots, P-1. && (8.59) \end{aligned}$$

As previously mentioned, we can derive the standard cost matrix from the monocular cost matrix. We can shift all terms in the scale column and row in the total cost matrix to the associated column and row of the homogenizing variable. Then, we delete the scale column and row from the total cost matrix.

8.2.6 Reducing the Dimensionality of the QCQP

Similar HE calibration, if we know the optimal rotation matrices $\mathbf{R}^\star_{\mathbf{X}_j}, \mathbf{R}^\star_{\mathbf{Y}_k}$ for $j = 0, \ldots, M-1$ and $k = 0, \ldots, P-1$, then the unconstrained optimal $\mathbf{X}$ translation vectors $\mathbf{t}^\star_{\mathbf{X}_j}$ for $j = 0, \ldots, M-1$, $\mathbf{Y}$ translation vectors $\mathbf{t}^\star_{\mathbf{Y}_k}$ for $k = 0, \ldots, P-1$, and scale $\alpha^\star$ can be recovered by solving the linear system induced by Equation (8.42):

$$\begin{bmatrix} \mathbf{t}^\star_{\mathbf{X}_{0:K}}{}^\mathsf{T} & \mathbf{t}^\star_{\mathbf{Y}_{0:L}}{}^\mathsf{T} & \alpha^\star \end{bmatrix} = -\mathbf{U}^{-1}\mathbf{V}\mathbf{r}^\star. \tag{8.60}$$

Using the Schur complement, we reduce the cost matrix to

$$\mathbf{Q}_{red} = \mathbf{W} - \mathbf{V}^\mathsf{T}\mathbf{U}^{-1}\mathbf{V}. \tag{8.61}$$

The problem is reduced to the following form.

Problem 21. *Reduced QCQP Formulation of Monocular HERW Calibration.*

$$\begin{aligned} \min_{\mathbf{r}_{\mathbf{X}_{0:M-1}}, \mathbf{r}_{\mathbf{Y}_{0:P-1}}, s} \; & \mathbf{r}^\mathsf{T}\mathbf{Q}_{red}\mathbf{r}, \\ \text{s.t. } & \mathbf{R}_{\mathbf{X}_j} \in \mathrm{SO}(3) \; \forall \; j = 0, \ldots, M-1, \\ & \mathbf{R}_{\mathbf{Y}_k} \in \mathrm{SO}(3) \; \forall \; k = 0, \ldots, P-1, \\ & s^2 = 1; \end{aligned} \tag{8.62}$$

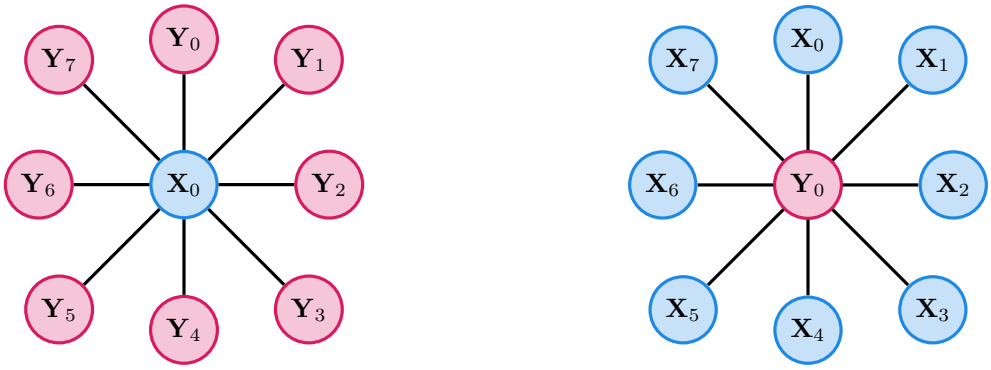
where

$$\mathbf{r}^\mathsf{T} = \begin{bmatrix} \mathbf{r}_{\mathbf{X}_0}{}^\mathsf{T} & \cdots & \mathbf{r}_{\mathbf{X}_{M-1}}{}^\mathsf{T} & \mathbf{r}_{\mathbf{Y}_0}{}^\mathsf{T} & \cdots & \mathbf{r}_{\mathbf{Y}_{P-1}}{}^\mathsf{T} & s \end{bmatrix}. \tag{8.63}$$

To convert the rotation matrix constraints into quadratic constraints, we leverage the homogenized constraints from Section 3.2.1.

Due to the general nature of the $\mathbf{AX} = \mathbf{YB}$ geometric constraint, we can interchange the transforms represented by $\mathbf{X}$ and $\mathbf{Y}$, provided that $\mathbf{B}$ is the only noisy pose measurement (see Section 8.2.2). The difference between state labelling choices becomes apparent after reducing the dimensionality of total cost matrix. Consider the two formulations shown in Figure 8.3. Both graphs could represent eight cameras observing an arm with a target of unknown scale. The fill patterns after performing the Schur complement are shown in Figure 8.4.

In contrast to the many $\mathbf{Y}$s case, we maintain a block arrowhead pattern in the many $\mathbf{X}$s formulation, which decreases the time required to solve the semidefinite relaxation of the QCQP. We recognize that some HERW calibration problems may be more complex than the cases in Figure 8.3 (i.e., involve many $\mathbf{X}$s and $\mathbf{Y}$s that are densely interconnected). However, we leave investigation of these cases for the future.



(a) A graph with many $\mathbf{Y}$s.

(b) A graph with many $\mathbf{X}$s.

Figure 8.3: Two potential formulations of the same scenario.

(a) Matrix sparsity pattern of a formulation with many $\mathbf{Y}$s.

(b) Matrix sparsity pattern of a formulation with many $\mathbf{X}$s.

Figure 8.4: Sparse matrix patterns for the many $\mathbf{X}$s and $\mathbf{Y}$s formulations. In these diagrams, the blue areas with no borders are the default fill patterns of the problems, while the red areas with dashed borders are the matrix areas that are filled by the schur complement dimensionality reduction. We would also like to note that the blue fill areas represent the general fill pattern of the sparse matrix and may have further sparsity patterns. The empty row and column represent the terms in the cost function with the homogenization term, which only exists in the constraints in the unscaled case.

8.3 Solving the Non-Convex QCQP

To start, we must compute the Lagrangian for Problem 21. In Problem 21, we must enforce $M + P$ sets of rotation matrix constraints. Using the homogenized constraints from Section 3.2.1, the Lagrangian function of Problem 21 is

$$L(\mathbf{x}, \boldsymbol{\nu}) = \nu_s + \mathbf{x}^T \mathbf{Z}(\boldsymbol{\nu}) \mathbf{x}, \tag{8.64}$$

$$\mathbf{Z}(\boldsymbol{\nu}) = \mathbf{Q}_{red} + \mathbf{P}_1(\boldsymbol{\nu}) + \mathbf{P}_2(\boldsymbol{\nu}), \tag{8.65}$$

where

$$\mathbf{P}_1(\boldsymbol{\mathcal{V}}_1,\ldots,\boldsymbol{\mathcal{V}}_{2(M+P)}) = \begin{bmatrix} -(\boldsymbol{\mathcal{V}}_1 \oplus \boldsymbol{\mathcal{V}}_2) & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \ddots & \ddots & \vdots \\ \vdots & \ddots & -(\boldsymbol{\mathcal{V}}_{2(M+P)-1} \oplus \boldsymbol{\mathcal{V}}_{2(M+P)}) & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \sum_{i=1}^{2(M+P)} \operatorname{tr}(\boldsymbol{\mathcal{V}}_i) \end{bmatrix}, \tag{8.66}$$

$$\mathbf{P}_2(\boldsymbol{\nu}_1,\ldots,\boldsymbol{\nu}_{(M+P)}) = \begin{bmatrix} \mathbf{P}_h(\boldsymbol{\nu}_1) & \mathbf{0} & \cdots & \mathbf{0} & -\mathbf{P}_o\boldsymbol{\nu}_1 \\ \mathbf{0} & \mathbf{P}_h(\boldsymbol{\nu}_2) & \ddots & \vdots & -\mathbf{P}_o\boldsymbol{\nu}_2 \\ \vdots & \ddots & \ddots & \mathbf{0} & \vdots \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{P}_h(\boldsymbol{\nu}_{(M+P)}) & -\mathbf{P}_o\boldsymbol{\nu}_{(M+P)} \\ -\boldsymbol{\nu}_1^\intercal\mathbf{P}_o^\intercal & -\boldsymbol{\nu}_2^\intercal\mathbf{P}_o^\intercal & \cdots & -\boldsymbol{\nu}_{(M+P)}^\intercal\mathbf{P}_o^\intercal & -\nu_s \end{bmatrix}, \tag{8.67}$$

$$\boldsymbol{\mathcal{V}}_1,\ldots,\boldsymbol{\mathcal{V}}_{2(M+P)} \in \mathbb{S}^3, \tag{8.68}$$

$$\boldsymbol{\nu}_1,\ldots,\boldsymbol{\nu}_{(M+P)} \in \mathbb{R}^9. \tag{8.69}$$

We can subdivide $\boldsymbol{\nu}_l$, such that $\boldsymbol{\nu}_l^\intercal = \begin{bmatrix} \boldsymbol{\nu}_{l,ijk}^T & \boldsymbol{\nu}_{l,jki}^T & \boldsymbol{\nu}_{l,kij}^T \end{bmatrix}$ and $\boldsymbol{\nu}_{l,ijk}, \boldsymbol{\nu}_{l,jki}, \boldsymbol{\nu}_{l,kij} \in \mathbb{R}^3 \; \forall \; l = 1,\ldots,M+P$. Further, the matrices $\mathbf{P}_o$ and $\mathbf{P}_h$ are

$$\mathbf{P}_o = \begin{bmatrix} \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} \\ \mathbf{I} & \mathbf{0} & \mathbf{0} \end{bmatrix}, \tag{8.70}$$

$$\mathbf{P}_h(\boldsymbol{\nu}_l) = \begin{bmatrix} \mathbf{0}_{3\times3} & -\boldsymbol{\nu}_{l,ijk}^\wedge & \boldsymbol{\nu}_{l,kij}^\wedge \\ \boldsymbol{\nu}_{l,ijk}^\wedge & \mathbf{0}_{3\times3} & -\boldsymbol{\nu}_{l,jki}^\wedge \\ -\boldsymbol{\nu}_{l,kij}^\wedge & \boldsymbol{\nu}_{l,jki}^\wedge & \mathbf{0}_{3\times3} \end{bmatrix}. \tag{8.71}$$

As discussed in Section 3.2.4, the minimum of the Lagrangian function is only defined if $\mathbf{Z}(\boldsymbol{\nu}) \succeq 0$. Consequently, the Lagrangian dual problem is as follows.

Problem 22 (Dual of HERW Calibration Problem)**.**

$$\begin{aligned} &\max_{\boldsymbol{\nu}} \; \nu_s, \\ &\text{s.t. } \mathbf{Z}(\boldsymbol{\nu}) \succeq 0, \end{aligned} \tag{8.72}$$

where $\mathbf{Z}(\boldsymbol{\nu})$ is defined in Equation (8.65) and $\boldsymbol{\nu} \in \mathbb{R}^{22(M+P)+1}$.

8.4 Experiments

We now report our experimental results. Section 8.4.1 presents two simulation studies, while Section 8.4.2 presents a real world experimental trial. In our simulation studies, we generated data for two different systems. We compared the estimation accuracy a variety of HERW and monocular HERW calibration schemes for these systems. Finally, in our real world experiment, we used HERW and monocular HERW calibration to estimate the transforms between eight cameras that were mounted on a mobile platform.

8.4.1 Simulation Studies

We verify the robustness of our algorithm to measurement noise in simulation, and compare the accuracy against variety of other HERW calibration methods. We simulated two robotic systems and performed two simulation studies on each system, which gives a total of four simulation studies. The first system consisted of a robotic manipulator with a wrist-mounted camera observing a visual fiducial target. Using the simulated manipulator wrist poses and camera-target measurements, this system formed a HERW calibration problem with one $\mathbf{X}$ and $\mathbf{Y}$. The second system generated data for a HERW calibration problem with four $\mathbf{X}$s and one $\mathbf{Y}$. The simulated system was a robotic manipulator with a wrist-mounted target that was observed by four stationary cameras. For each system, we generated data to study HERW and monocular HERW calibration.

The data generation process was similar for both simulated robotic systems. To generate data for the robotic manipulator with a wrist-mounted camera, we computed a camera trajectory relative to the target and chose groundtruth values for $\mathbf{X}$ and $\mathbf{Y}$. At each time t_i, we used the camera trajectory relative to the target to determine the camera-target transform, $\mathbf{B}_i$. By combining the ground truth values of $\mathbf{X}$, $\mathbf{Y}$, and $\mathbf{B}_i$, we computed the ground truth values for each $\mathbf{A}_i$. Similarly, we generated data for the cameras observing a robotic manipulator with a wrist-mounted target simulation by defining the target trajectory relative to the stationary cameras. The target trajectory enabled computation of the $\mathbf{B}_{ij}$ measurements, where t_i was the measurement time for the jth camera. Using the ground truth values for $\mathbf{X}_j$, $\mathbf{Y}$, and $\mathbf{B}_{ij}$, we computed the ground truth measurements for each $\mathbf{A}_{ij}$. We added zero-mean Gaussian noise, $\mathbf{n}_t \sim \mathcal{N}\left(\mathbf{0}, \sigma_t^2 \mathbf{I}\right)$, to the ground truth $\mathbf{t}_{\mathbf{B}_i}$ and $\mathbf{t}_{\mathbf{B}_{ij}}$ measurements. Further, we right-perturbed the noiseless rotation measurements, $\mathbf{R}_{\mathbf{B}_i}$ and $\mathbf{R}_{\mathbf{B}_{ij}}$, with samples from $\mathbf{R}_n \sim \text{Lang}\left(\mathbf{I}, \kappa\right)$. During each study, we varied the camera rotation measurement noise levels between $\kappa = 125$ and $\kappa = 12$, while the translation noise levels were varied between $\sigma_t = 1$ cm and $\sigma_t = 5$ cm. In the monocular HERW calibration studies, we scaled the translation component of the noisy $\mathbf{B}_i$ measurements by 0.5. Finally, we generated 100 runs for each study.

Using the data from our simulated studies, we compared the estimated parameter accuracy of our HERW and monocular HERW methods to five HERW methods and one monocular HERW method. The five comparison HERW solvers are the two-stage closed-form methods in Shah [2013] and Wang et al. [2022], certifiable method in Horn et al. [2023], probabilistic method in Dornaika and Horaud [1998], and an on-manifold (i.e., local), probabilistic solver (see Appendix B for our implementation). For our monocular HERW calibration scenarios, we compared the accuracy of our algorithm to the local probabilistic solver. When the scenario had more than one $\mathbf{X}$ to solve for, we compared our methods to Wang et al. [2022], Horn et al. [2023], and the local solver.

For the solvers that required initialization, we provided random initial values, the solution from Wang et al. [2022], or parameters close to the ground truth. For every study, our method and the method in Horn et al. [2023] were randomly initialized. We attempted randomly initializing the local solver, but this scheme resulted in the local solver returning local minima. Consequently, we initialized the Dornaika and Horaud [1998] solver and our local solver using the solution from the method in Wang et al. [2022]. However, the method in Wang et al. [2022] cannot solve monocular HERW calibration problems. In the monocular HERW studies, we initialized the local solver with calibration parameters that were within 10 cm and 10° of the ground truth values. Additionally, we set the initial scale to 1.

Robot Arm Poses on a Sphere

In this pair of HERW and monocular HERW studies, we used the problem of extrinsically calibrating a wrist-mounted camera and a target, relative to a robotic manipulator, as a setting to compare the robustness of HERW

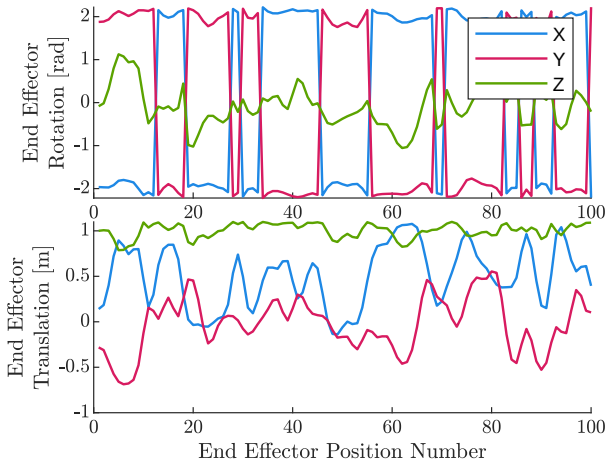


Figure 8.5: Plot of end effector pose relative to the base for the robot poses on sphere experiment with known scale. We treat the data as if the robot stops at each position.

and monocular HERW solvers to measurement noise. A diagram of the simulated system is shown in Figure 8.1. Importantly, the exact robot arm in this diagram is not simulated or necessarily representative of the robot arm that could perform this task. In particular, we estimated $\mathbf{X} = \mathbf{T}_{hc}$ and $\mathbf{Y} = \mathbf{T}_{bw}$, where $\underrightarrow{\mathcal{F}}_b$, $\underrightarrow{\mathcal{F}}_w$, $\underrightarrow{\mathcal{F}}_h$, $\underrightarrow{\mathcal{F}}_c$, are the manipulator base, target, wrist, and camera reference frames, respectively. For each data collection run, the camera travelled along a sphere of radius 0.5 m, while the z-axis of the camera pointed towards the center of the target. The camera trajectory was different for each run and consisted of 100 manipulator positions. A representative wrist trajectory, which is derived from the camera trajectory, is shown in Figure 8.5. The trajectories generated for the HERW calibration study were not identifiable for the monocular HERW problem. To create identifiable runs for the monocular HERW problem, the camera travelled on a sphere of radius 0.25 m after the 50th manipulator position (see Figure 8.6).

For the HERW calibration study, the mean and standard deviation of the estimated translation ($t_{x,err}$,$t_{y,err}$) and rotation ($r_{x,err}$,$r_{y,err}$) errors are shown in Table 8.2. Generally, the least accurate methods were the two-stage closed-form solvers from Shah [2013] and Wang et al. [2022], while our method and the local solver were the most accurate. Often, our and the local method returned similar solutions, which outperformed the other methods by up to 50 mm and 3 deg. However, our and the local approaches were more susceptible to translation

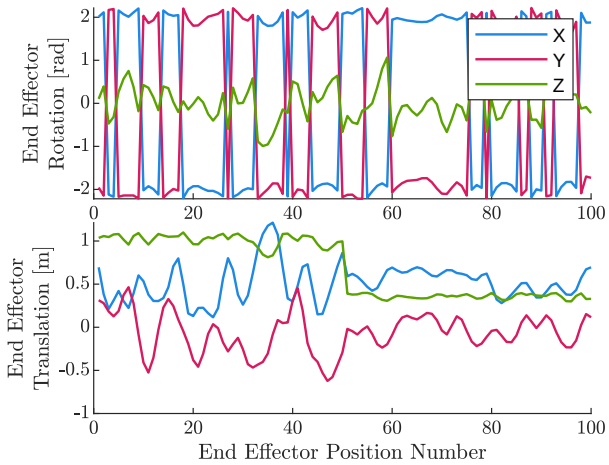


Figure 8.6: Plot of end effector pose relative to the base for the robot poses on sphere experiment with unknown scale. We treat the data as if the robot stops at each position.

Table 8.2: Calibration results for the robot_world_poses_on_sphere dataset. The values in each row are estimated by a different algorithm. The mean error magnitude and standard deviation are given in each cell. Dornaika and the local solvers are initialized with the results from Wang. Initializing the local method with random poses results in no correct solutions.

Noise Level	Method (Init.)	$t_{x,err}$ [mm]	$r_{x,err}$ [deg]	$t_{y,err}$ [mm]	$r_{y,err}$ [deg]
$\kappa = 125$, $\sigma_t = 1$cm	Shah	20.6 ± 10.5	1.37 ± 0.55	9.9 ± 4.8	1.34 ± 0.57
	Wang	20.6 ± 10.5	1.37 ± 0.55	9.9 ± 4.8	1.34 ± 0.57
	Dornaika (Wang)	19.9 ± 10.2	1.25 ± 0.53	8.8 ± 4.3	1.20 ± 0.54
	Horn	18.5 ± 9.4	1.16 ± 0.51	8.4 ± 4.2	1.10 ± 0.51
	Local (Wang)	11.1 ± 5.82	0.78 ± 0.37	3.69 ± 2.12	0.63 ± 0.38
	Ours	10.9 ± 5.6	0.77 ± 0.36	3.71 ± 2.13	0.62 ± 0.36
$\kappa = 125$, $\sigma_t = 5$cm	Shah	31.2 ± 15.0	1.61 ± 0.64	21.2 ± 10.0	1.57 ± 0.66
	Wang	31.2 ± 15.0	1.61 ± 0.64	21.2 ± 10.0	1.57 ± 0.66
	Dornaika (Wang)	30.6 ± 14.5	1.49 ± 0.62	20.4 ± 9.7	1.44 ± 0.63
	Horn	29.1 ± 13.6	1.42 ± 0.60	19.9 ± 9.2	1.36 ± 0.59
	Local (Wang)	28.5 ± 13.4	1.45 ± 0.62	18.5 ± 8.8	1.39 ± 0.62
	Ours	28.4 ± 13.0	1.42 ± 0.63	18.5 ± 8.7	1.36 ± 0.61
$\kappa = 12$, $\sigma_t = 1$cm	Shah	65.5 ± 38.3	4.34 ± 1.95	31.8 ± 17.2	4.41 ± 1.98
	Wang	65.5 ± 38.3	4.34 ± 1.95	31.8 ± 17.2	4.41 ± 1.98
	Dornaika (Wang)	63.4 ± 37.3	3.92 ± 1.87	27.4 ± 16.2	3.92 ± 1.90
	Horn	45.0 ± 24.8	3.30 ± 1.69	25.8 ± 14.1	3.19 ± 1.55
	Local (Wang)	15.2 ± 10.0	1.84 ± 0.84	3.4 ± 1.8	0.88 ± 0.59
	Ours	15.1 ± 10.0	1.81 ± 0.83	3.4 ± 1.8	0.87 ± 0.59
$\kappa = 12$, $\sigma_t = 5$cm	Shah	71.9 ± 44.7	4.74 ± 2.13	37.8 ± 19.3	4.63 ± 2.21
	Wang	71.9 ± 44.7	4.74 ± 2.13	37.8 ± 19.3	4.63 ± 2.21
	Dornaika (Wang)	69.9 ± 43.1	4.33 ± 2.08	33.9 ± 17.9	4.13 ± 2.14
	Horn	50.6 ± 28.4	3.69 ± 1.72	31.0 ± 16.2	3.40 ± 1.72
	Local (Wang)	48.3 ± 27.3	3.18 ± 1.67	18.7 ± 9.04	2.72 ± 1.64
	Ours	47.7 ± 27.0	3.12 ± 1.67	18.8 ± 9.02	2.68 ± 1.65

measurement noise than rotation measurement noise. Finally, all algorithms have roughly the same accuracy when $\kappa = 125$ and $\sigma_t = 5$ cm. In this case, the rotation measurement noise is low, so the two-stage closed-form solvers are likely to return accurate rotation estimates. Further, the contribution of the rotation error distributions to the final cost of the optimization problem is small relative to the translation error distributions, so ignoring the relative distributions of the rotation and translation errors has a small impact on estimation accuracy.

Table 8.3 contains the mean and standard deviation of the estimated translation ($t_{x,err}$,$t_{y,err}$), rotation ($r_{x,err}$,$r_{y,err}$), and scale α_{err} error for the monocular HERW study. Similar to the HERW study, our method and the local method had the same estimation accuracy. However, our method did not require an initial estimate of $\mathbf{X}$, $\mathbf{Y}$, or α of the system. Eventhough monocular HERW calibration was a harder problem than HERW calibration, both monocular HERW solvers returned more accurate estimates than the corresponding HERW calibration methods. We posit, based on our experiments, that this change of estimation accuracy was due to constraining the camera trajectory to a single sphere in the HERW calibration study. Collecting data on the surface of two spheres may have resulted in more "identifiable" data, which improved the accuracy of the estimated parameters (see Grebe [2022] for more insight into this topic).

Many Cameras Observing a Wrist-Mounted Target

In this study, we compared the estimated parameter accuracy of HERW and monocular HERW algorithms for systems with multiple $\mathbf{X}$s and one $\mathbf{Y}$. The simulated system consisted of a robotic manipulator with a wrist-mounted target that was viewed by four fixed cameras. Figure 8.7 shows a diagram of the base-camera and wrist-target transforms for one camera. Each estimated $\mathbf{X}_i = \mathbf{T}_{bc_i}$, where $\underrightarrow{\mathcal{F}}_b$ and $\underrightarrow{\mathcal{F}}_{c_i}$ were the manipulator base and ith camera reference frames, respectively. The transform from the robot wrist to the target was $\mathbf{Y} = \mathbf{T}_{ht}$, where $\underrightarrow{\mathcal{F}}_h$ and $\underrightarrow{\mathcal{F}}_t$ are the manipulator wrist and target reference frames, respectively. Each trajectory consisted of 108 manipulator positions. While each data collection run had a different trajectory, the trajectory of the target

Table 8.3: Calibration results for the robot_world_poses_on_sphere_scale dataset. The values in each row are estimated by a different algorithm. The mean error magnitude and standard deviation are given in each cell. The local method is initialized with parameters 10° and 10 cm from the ground truth values and a scale of 1. Random pose initialization of the local solver leads to no correct solutions.

Noise Level	Method (Init.)	$t_{x,err}$ [mm]	$r_{x,err}$ [deg]	$t_{y,err}$ [mm]	$r_{y,err}$ [deg]	α_{err} [%]
$\kappa = 125$, $\sigma_t = 1$cm	Local (close)	4.61 ± 2.39	0.51 ± 0.21	3.82 ± 1.93	0.22 ± 0.12	1.78e-3 ± 1.54e-3
	Ours	4.71 ± 2.43	0.513 ± 0.218	3.82 ± 1.91	0.250 ± 0.113	1.76e-3 ± 1.54e-3
$\kappa = 125$, $\sigma_t = 5$cm	Local (close)	23.0 ± 11.2	0.94 ± 0.44	19.1 ± 11.4	0.85 ± 0.40	8.45e-3 ± 7.03e-3
	Ours	23.8 ± 11.3	1.11 ± 0.53	19.0 ± 11.4	1.06 ± 0.52	8.41e-3 ± 7.02e-3
$\kappa = 12$, $\sigma_t = 1$cm	Local (close)	4.52 ± 1.97	1.57 ± 0.59	3.78 ± 1.89	0.20 ± 0.10	1.88e-3 ± 1.37e-3
	Ours	4.50 ± 1.99	1.54 ± 0.57	3.77 ± 1.88	0.206 ± 0.098	1.87e-3 ± 1.37e-3
$\kappa = 12$, $\sigma_t = 5$cm	Local (close)	23.2 ± 14.1	1.90 ± 0.81	20.0 ± 11.0	1.07 ± 0.54	9.24e-3 ± 7.69e-3
	Ours	24.4 ± 14.3	1.99 ± 0.81	19.6 ± 11.0	1.25 ± 0.58	9.16e-3 ± 7.69e-3

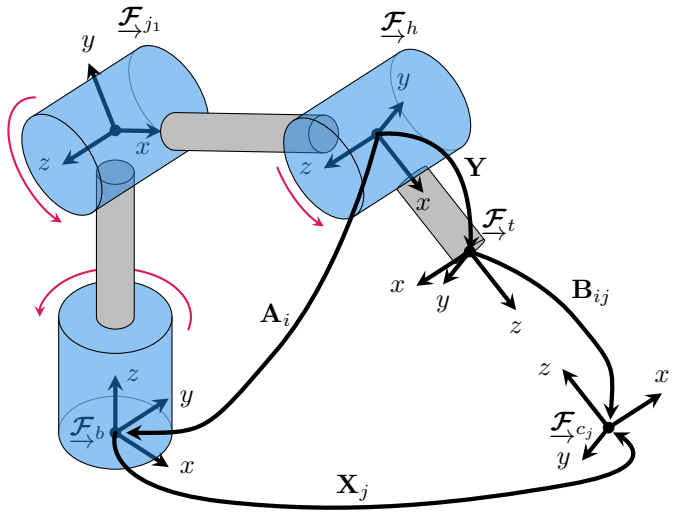


Figure 8.7: A 2D view of a robot arm with a wrist mounted target that is observed by a stationary camera. In this diagram, the base, joint 1, wrist, camera j, and target reference frames are labelled $\underrightarrow{\mathcal{F}}_b$, $\underrightarrow{\mathcal{F}}_{j_1}$, $\underrightarrow{\mathcal{F}}_h$, $\underrightarrow{\mathcal{F}}_{c_j}$, and $\underrightarrow{\mathcal{F}}_t$, respectively. The red arrows indicate the joint rotation axis. At time t_i, we use the forward kinematics of the manipulator to estimate the transformation from the manipulator base to the wrist, $\mathbf{A}_i$. Further, we can measure the transform from the target to camera j, $\mathbf{B}_{ij}$.

for one run is shown in Figure 8.8. We designed the trajectories such that the target was always visible to every fixed camera and both the HERW and monocular HERW problems were identifiable.

The results of HERW and monocular HERW studies are shown in Tables 8.4 and 8.5. Tables 8.4 and 8.5 contain the mean and standard deviation of the estimated translation ($t_{x,err}$,$t_{y,err}$), rotation ($r_{x,err}$,$r_{y,err}$) error. The errors associated with $\mathbf{X}$ (i.e., $t_{x,err}$ and $r_{x,err}$) are the mean and standard deviation of all four estimated $\mathbf{X}$ transforms. Table 8.5 also has the mean and standard deviation of the estimated scale error, α_{err}. Similar to the previous studies, our method and the local method returned the most accurate estimated parameters. Despite the method in Wang et al. [2022] projecting the solution onto $\mathrm{SO}(3)$, the method in Wang et al. [2022] was more accurate than the method in Horn et al. [2023]. Similar to our previous monocular HERW study, ours and the local method achieved the same estimated parameter accuracy.

8.4.2 Real World Experiment

Herein, we discuss our data collection system, data preprocessing procedure, and calibration results.

Table 8.4: Calibration results for the fixed_camera_robot_world dataset. The values in each row are estimated by a different algorithm. The mean error magnitude and standard deviation are given in each cell. The local solver is initialized with the results from Wang. Note we have no apriori estimates of $\mathbf{t}_{\mathbf{X}_i}$. Initializing the local solver with random poses leads to no correct solutions.

Noise Level	Method (Init.)	$t_{x,err}$ [mm]	$r_{x,err}$ [deg]	$t_{y,err}$ [mm]	$r_{y,err}$ [deg]
$\kappa = 125$, $\sigma_t = 1$cm	Wang	2.32 ± 1.56	0.469 ± 0.200	2.71 ± 1.37	0.273 ± 0.120
	Horn	12.11 ± 6.49	0.455 ± 0.191	5.43 ± 2.04	0.165 ± 0.067
	Local (Wang)	0.994 ± 0.416	0.454 ± 0.190	0.572 ± 0.296	0.022 ± 0.010
	Ours	0.992 ± 0.416	0.454 ± 0.189	0.570 ± 0.294	0.021 ± 0.010
$\kappa = 125$, $\sigma_t = 5$cm	Wang	4.95 ± 2.20	0.470 ± 0.196	3.52 ± 1.57	0.260 ± 0.115
	Horn	12.4 ± 6.4	4.55 ± 1.91	5.42 ± 2.33	1.76 ± 0.69
	Local (Wang)	4.57 ± 1.96	0.460 ± 0.191	2.74 ± 1.34	0.120 ± 0.056
	Ours	4.56 ± 1.96	0.459 ± 0.191	2.69 ± 1.32	0.111 ± 0.051
$\kappa = 12$, $\sigma_t = 1$cm	Wang	6.05 ± 4.84	1.54 ± 0.63	7.63 ± 4.04	0.812 ± 0.364
	Horn	69 ± 121	24.6 ± 94.4	48.5 ± 82.5	13.0 ± 76.8
	Local (Wang)	0.933 ± 0.370	1.52 ± 0.64	0.537 ± 0.262	0.021 ± 0.010
	Ours	0.933 ± 0.370	1.50 ± 0.62	0.537 ± 0.262	0.021 ± 0.010
$\kappa = 12$, $\sigma_t = 5$cm	Wang	7.85 ± 4.34	1.52 ± 0.70	9.58 ± 4.46	0.868 ± 0.340
	Horn	77 ± 146	26 ± 106	57 ± 110	14.8 ± 83.9
	Local (Wang)	4.57 ± 1.97	1.51 ± 0.682	2.93 ± 1.34	0.104 ± 0.047
	Ours	4.57 ± 1.97	1.48 ± 0.67	2.92 ± 1.32	0.103 ± 0.046

Table 8.5: Calibration results for the fixed_camera_robot_world_scale dataset. The values in each row are estimated by a different algorithm. The mean error magnitude and standard deviation are given in each cell. The local method is initialized with parameters 10° and 10 cm from the ground truth values and a scale of 1. Random pose initialization of the local solver leads to no correct solutions.

Noise Level	Method (Init.)	$t_{x,err}$ [mm]	$r_{x,err}$ [deg]	$t_{y,err}$ [mm]	$r_{y,err}$ [deg]	α_{err} [%]
$\kappa = 125$, $\sigma_t = 1$cm	Local (close)	1.05 ± 0.45	0.469 ± 0.185	0.631 ± 0.249	0.030 ± 0.013	2.71e-4 ± 1.98e-4
	Ours	1.05 ± 0.45	0.468 ± 0.185	0.597 ± 0.256	0.026 ± 0.012	2.71e-4 ± 1.98e-4
$\kappa = 125$, $\sigma_t = 5$cm	Local (close)	5.25 ± 2.20	0.457 ± 0.195	3.15 ± 1.26	0.179 ± 0.079	1.37e-3 ± 1.03e-3
	Ours	5.21 ± 2.18	0.454 ± 0.194	3.03 ± 1.24	0.159 ± 0.071	1.37e-3 ± 1.03e-3
$\kappa = 12$, $\sigma_t = 1$cm	Local (close)	1.02 ± 0.44	1.50 ± 0.62	0.530 ± 0.284	0.021 ± 0.010	2.58e-4 ± 2.02e-4
	Ours	1.02 ± 0.44	1.46 ± 0.61	0.530 ± 0.284	0.021 ± 0.010	2.57e-4 ± 2.02e-4
$\kappa = 12$, $\sigma_t = 5$cm	Local (close)	5.20 ± 2.27	1.48 ± 0.64	3.13 ± 1.56	0.156 ± 0.075	1.18e-3 ± 8.63e-4
	Ours	5.18 ± 2.26	1.44 ± 0.626	3.02 ± 1.51	0.144 ± 0.069	1.18e-3 ± 8.62e-4

Data Collection and Data Preprocessing

In our real world experiment, we estimate the relative transforms between multiple cameras that are mounted on a mobile system. Generally, for systems with multiple cameras, we cannot place an AprilTag grid in the

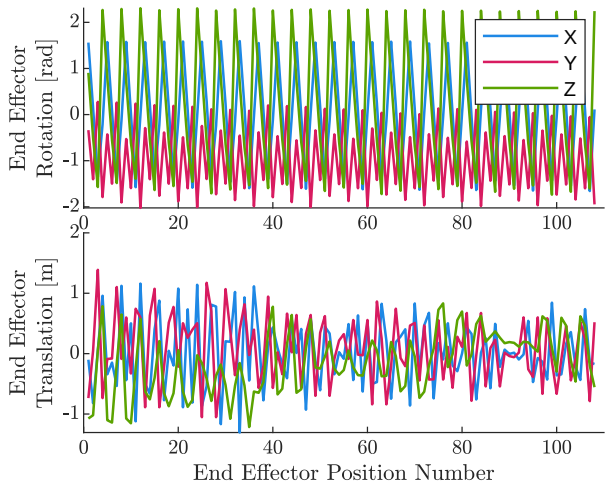


Figure 8.8: Motion for HERW simulation experiment. Note that this is for both experiments.

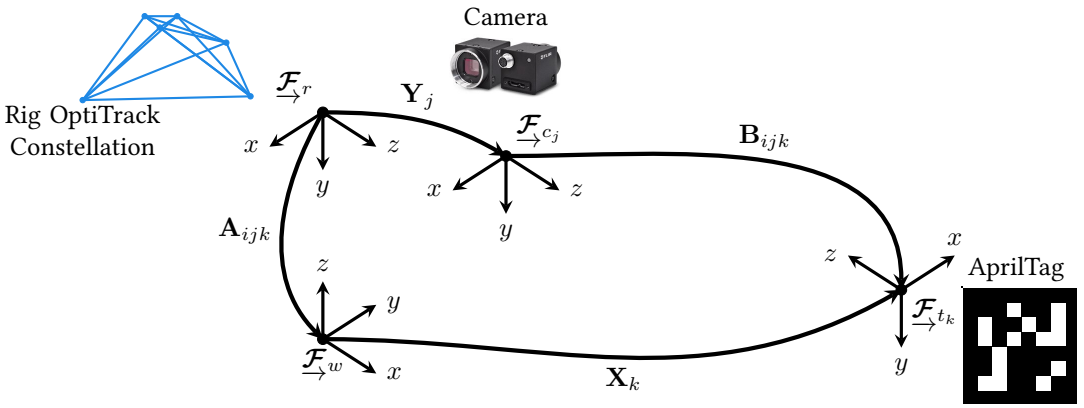


Figure 8.9: A diagram of the measurements for camera i and target j at time t_i. The reference frames for the camera, target, OptiTrack world, and rig reference frames are $\underrightarrow{\mathcal{F}}_{c_j}$, $\underrightarrow{\mathcal{F}}_{t_k}$, $\underrightarrow{\mathcal{F}}_w$, and $\underrightarrow{\mathcal{F}}_r$. The OptiTrack rig constellation enables estimation of the transform $\mathbf{A}_{ijk}$. The monocular camera observing the the AprilTag enables estimation of the transform $\mathbf{B}_{ijk}$.

environment such that all cameras simultaneously view the grid. Since the method in Rehder et al. [2016a] relies on this property to extrinsically calibrate the cameras, we must perform multiple data collection runs, which is a tedious process. However, we can modify this calibration problem to take the form of a HERW calibration problem and perform one data collection run. If we place OptiTrack markers on the mobile system and AprilTags in the environment, then we enable the HERW calibration scheme shown in Figure 8.9. By moving the mobile system, we can provide sufficient excitation to make the HERW calibration problem identifiable.

Figures 8.10 and 8.11 show our mobile system and two images of our experimental environment, respectively. The sensors on our mobile system consist of eight Point Grey Blackfly S USB cameras, OptiTrack markers, and a VectorNav vn-100 IMU. We placed a sufficient number of OptiTrack markers on the system to enable estimation of the relative transform between the OptiTrack reference frame and a reference frame that is fixed to the mobile system. To validate the camera calibration parameters estimated using HERW calibration, we leveraged the IMU mounted on the system to determine the ground truth camera transforms using Rehder et al. [2016a]. Eight 15.1 cm and eight 16.2 cm AprilTags, from the tag family tag36h11, were mounted in the experimental environment, for a total of sixteen AprilTags. To evaluate the accuracy of our estimated AprilTag parameters, we enabled ground truth measurement of an AprilTag by placing OptiTrack markers on the AprilTag with id code 20.

Figure 8.12 shows the trajectory of the mobile system. In the first half of the data collection run, the system experienced purely planar motion, which ensured that each target was observed at least once. After the planar motion, the system rotated about all three axes and translated perpendicular to the plane of motion. This component of the trajectory ensured sufficient excitation for the HERW calibration problem. Figure 8.13 is a grid where each row is a camera and each column is a target. If the grid square for a column and row are blue, then the data collected for that camera-target pair enables an identifiable HERW calibration subproblem. A red square indicates that the data collected for a camera-target pair did not contain sufficient excitation, while a white square indicates the camera did not observe the target. From Figure 8.13, each camera had at least one target that enabled a HERW calibration subproblem. However, the data associated AprilTags 6, 8, and 23 did not contain sufficient excitation and subsequently cannot be estimated using data from a single camera-target pair.

Our data preprocessing pipeline for this experiment consisted of four steps. First, we rectified the images using the camera intrinsic parameters that were estimated using Kalibr (see Rehder et al. [2016a]). Second, we measured the camera to AprilTag transforms using an AprilTag detector. Third, we synchronized the camera

and OptiTrack measurements by linearly interpolating the OptiTrack system measurements to the camera measurement time stamps. Finally, we used a RANSAC pose averaging scheme to reject gross outliers. In the HERW case, if each $\mathbf{Y}_k$ is approximately known, then the HERW geometric constraint at time t_i becomes

$$\mathbf{X}_j = {\mathbf{A}_{ijk}}^{-1}\mathbf{Y}_k\mathbf{B}_{ijk}, \tag{8.73}$$

which is a pose averaging model. Consequently, we can use RANSAC in conjunction with the pose averaging model to determine the $\mathbf{A}_{ijk}$-$\mathbf{B}_{ijk}$ pairs that result in a consistent $\mathbf{X}_j$ transform. For our RANSAC procedure, our minimum inlier set size was one third of the number of $\mathbf{A}_{ijk}$-$\mathbf{B}_{ijk}$ pairs for camera j and target k. Further, data from camera-target pairs with only one $\mathbf{A}_{ijk}$-$\mathbf{B}_{ijk}$ pair were rejected, because they could not be validated using this scheme. An $\mathbf{A}_{ijk}$-$\mathbf{B}_{ijk}$ pair was an inlier if it was within 0.6 m and 60° of the estimated $\mathbf{X}_j$ transform. The inlier $\mathbf{A}_{ijk}$-$\mathbf{B}_{ijk}$ pairs were then saved for our HERW calibration problem. To extend this outlier rejection scheme to monocular HERW calibration, we assume that the AprilTag size is known within 10% of the actual value, which we empirically found to be sufficient for outlier rejection.

Experimental Results

Using the preprocessed data, we achieved the calibration results shown in Table 8.6, where the ground truth calibration values were determined using Rehder et al. [2016a]. We assumed that our hand-measured AprilTag sizes were approximately correct, so the local HERW and monocular HERW solvers were initialized with the parameters estimated using the method in Wang et al. [2022]. Our estimated AprilTag translation was within 12 cm (or 8% of the ground-truth distance) and 6° of the OptiTrack measured transform. The estimated camera calibration parameters were, on average, around 3 cm and 1° of the parameters estimated by Kalibr. We did not expect our algorithm to return the same values as Kalibr because collecting a dedicated calibration dataset for each camera should result in more accurate calibration parameters. As expected, the method in Wang et al. [2022] returned the least accurate rotation estimates since the estimated rotations were projected onto $\mathrm{SO}(3)$.

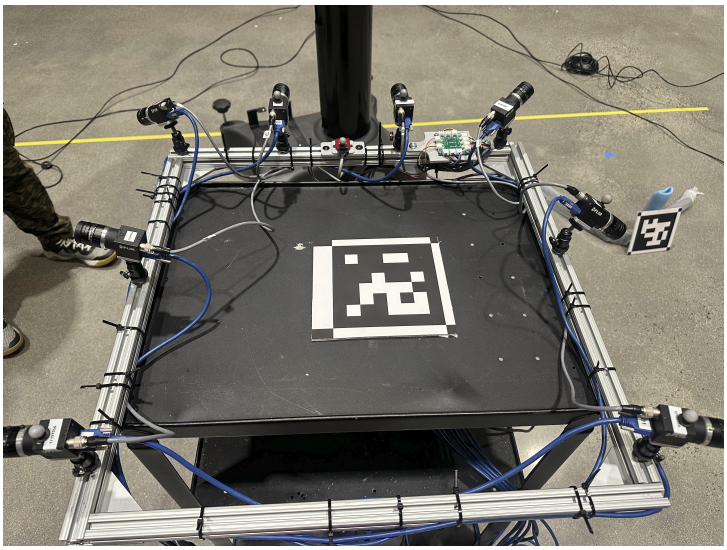


Figure 8.10: Image of the real-world data collection rig. The data collection rig consists of eight hardware synchronized cameras facing a variety of different directions. Further, the data collection rig includes an IMU and opti track markers. The OptiTrack markers enable us to estimate the rig pose relative to the OptiTrack reference frame.

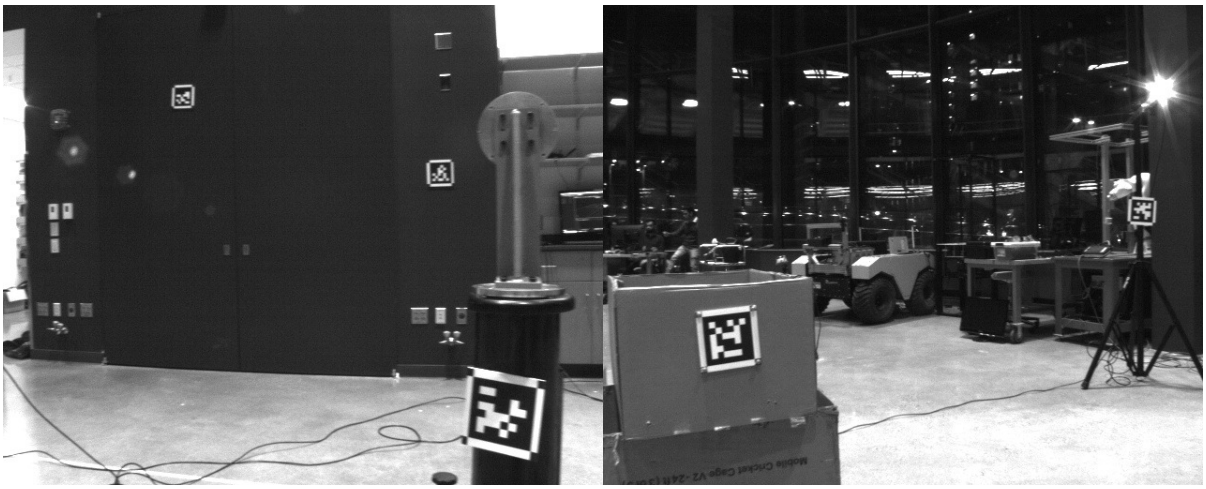


Figure 8.11: Images from the real-world experiment. These images are from camera 0 and show a subset of the apriltags in the environment. The bottom left AprilTag in the image on the right has OptiTrack markers, so we can determine the ground truth pose of the AprilTag frame relative to the OptiTrack world frame.

Table 8.6: Calibration results for our real world dataset with known scale. The values in each row are estimated by a different algorithm. The mean error magnitude and standard deviation are given in each cells corresponding to the $\mathbf{t}_{c_0}^{c_i c_0}$ and $\mathbf{R}_{c_0 c_i}$ errors. The error in the estimated transform between AprilTag 20 the OptiTrack reference frame is also provided. The local method is initialized with the solution from Wang et al. [2022].

Method (Init.)	$\mathbf{t}_{c_0}^{c_i c_0}$ Error [cm]	$\mathbf{R}_{c_0 c_i}$ Error [deg]	$\mathbf{t}_{w}^{t_{20} w}$ Error [cm]	$\mathbf{R}_{w t_{20}}$ Error [deg]
Wang et al. [2022]	3.46 ± 1.69	1.38 ± 0.81	11.7	5.30
Local (Wang)	3.34 ± 2.44	0.88 ± 0.49	11.4	4.50
Ours	3.20 ± 1.88	0.99 ± 0.52	11.5	4.73
Local with Scale (Wang)	3.28 ± 2.42	0.86 ± 0.47	7.92	4.50
Ours with Scale	2.87 ± 1.94	0.99 ± 0.52	7.94	4.72

Similar to our simulation studies, our method and the local method resulted in estimated parameters with approximately the same accuracy. Interestingly, estimating the scale of the AprilTags improved the estimated parameter accuracy. The estimated AprilTag scale was 2.5% smaller than the hand-measured value. From our

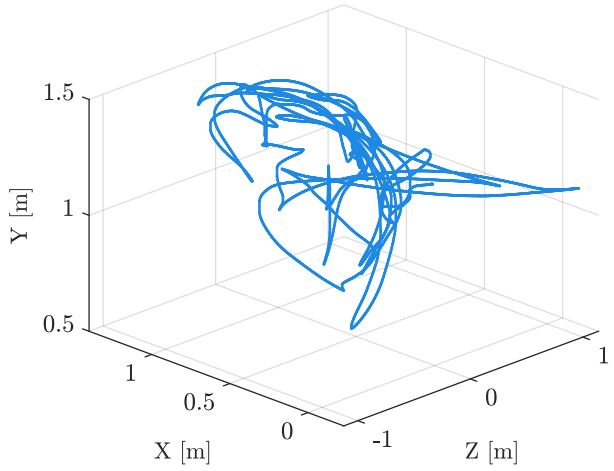


Figure 8.12: Trajectory of the rig in the real-world experiment. Initially, the platform rotates about the y-axis follows a planar motion in the xz-plane. Following the planar motion, the system follows an unconstrained trajectory, which allows for rotation about the x and z axes.

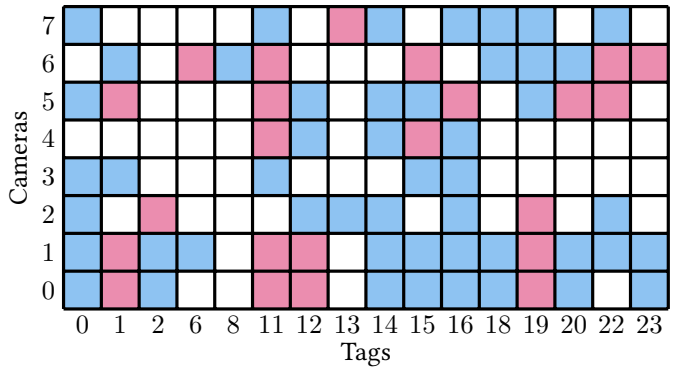


Figure 8.13: A grid describing the observability of each connection in the bipartite graph generated by our real-world experiment. Blue squares indicate that measurements between tag i and camera j are sufficient for an identifiable HERW problem. Red squares indicate that there is a connection between tag i and tag j, but the measurements are insufficient for the problem to be identifiable on its own. The data for subproblem i, j is insufficient if the measurements do not capture rotation about two axes. A white square indicates that there is no connection between tag i and camera j in the real-world problem.

simulation studies, the estimated scale error is often within 0.01% of ground truth value, so a scale error this large is unexpected. However, this scaling suggests that the hand measurements of the AprilTags were incorrect by approximately 3 mm, which is a reasonable. This correction improved our estimated camera calibration parameters and AprilTag translations by approximately 0.5 cm and 4 cm, respectively. As a result, even if a user hand-measures the AprilTag size there may be a benefit in estimating the scale.

8.5 Summary and Future Work

In this chapter, we presented a probabilistic, certifiable solver for the HERW and monocular HERW calibration problems. We demonstrated that our HERW and monocular HERW calibration problems were MLE problems. Further, we extended our solvers to handle multiple $\mathbf{X}$s and $\mathbf{Y}$s. We performed four simulation studies and a real world experiment. Our simulation studies demonstrated that our method is more robust to measurement noise than other certifiable methods. Additionally, our method returned similar solutions as an on-manifold approach without requiring initialization. Our real world experiments showed that the accuracy of our method is similar to competing approaches.

Potential avenues of future research involve making our method robust to outliers, exploring different HERW formulations, and sparisfying the bipartite graph. To reject outliers in our HERW and monocular HERW algorithms, we required a rough approximation of the solution. We could employ the binary cloning method from Yang et al. [2021] to fold the outlier rejection component into our optimization problem. As shown in this chapter, there are many potential formulations of the HERW calibration problem and there is no definitive choice on which set of sensors are $\mathbf{X}$ and $\mathbf{Y}$. Our choice of formulation is driven by the assumption that only sensor b is noisy, but there may be other noise models that remove this assumption. Empirically, there are some cases where a larger decrease in computation time may be achieved by relying on the chordal sparsity of the problem (see Garstka et al. [2021]). For our calibration problems, the maximum number of transforms that we estimated was 24. The potential number of transforms could increase drastically, so future work could extend methods, such as Doherty et al. [2022], to sparsify the connected bipartite graphs and speed up the optimization time.

Chapter 9

Conclusion

In this thesis, we explored four novel calibration methods. Our two radar calibration algorithms focused on removing the specialized target requirement by leveraging the ego-velocity estimates to sidestep the cross-modal data association problem. By eliminating the need for a radar target, we expanded possible calibration venues and sensor configurations. Our two certifiable calibration algorithms enabled us to guarantee that our solver returned the optimal calibration parameters for a given dataset. In this chapter, we conclude the thesis with a summary of our contributions and potential future research directions.

9.1 Summary of Contributions

We contributed four novel algorithms to the field of sensor calibration. Two of the novel algorithms calibrated sensor pairs in which at least one sensor was a radar. In particular, we developed the first spatiotemporal calibration algorithm for radar-camera pairs that does not require specialized targets. We demonstrated that ego-velocity estimates enable estimation of the yaw angle and direction of the translation vector between pairs of coplanar 2D radar sensors. Further, we determined that the full spatial transform between the coplanar 2D radars can be recovered when an additional, coarse source of rotational velocity information is available. For both of these calibration scenarios, we proved the observability or identifiability of the calibration problem and derived system excitation requirements. The second pair of novel algorithms focused on certifiably optimal calibration. We extended the certifiably optimal hand-eye calibration method in Giamou et al. [2019] to monocular hand-eye calibration. Finally, we developed the first certifiably optimal probabilistic HERW calibration method that can handle HERW calibration problems with many $\mathbf{X}$s and $\mathbf{Y}$s.

9.2 Future Research Directions

Our radar sensor calibration algorithms primarily leveraged the ego-velocity estimates of the radar. Future research could investigate alternative cost functions that explicitly consider alignment errors (similar to Peršić et al. [2021b]) and ego-velocity estimates. For systems with multiple radars, we could integrate the unbiased ego-velocity and biased rotational velocity estimates (i.e., the rotational velocity is scaled) to form a biased discrete motion model - similar to Forster et al. [2017]. The biased discrete motion model could simplify radar data association between measurements at different times.

Our certifiable sensor calibration algorithms are sensitive to outliers in the data and require synchronized

measurements. We could extend our method using the robust problem formulation from Hu et al. [2019]. This approach could increase the number of estimated transforms, so future work could require graph sparsification techniques, such as Doherty et al. [2022], to speed up the optimization time. While the challenge of accurate joint spatiotemporal extrinsic calibration has been explored by Rehder et al. [2016a], Lambert et al. [2016], Marr and Kelly [2018], and Furrer et al. [2018], a *certifiable* algorithm has not, to our knowledge, been proposed.

While this thesis does not include machine learning, our optimization problems can serve as differentiable layers in a neural network, as shown in Agrawal et al. [2019] and Pineda et al. [2022]. Consequently, we can learn certain challenging components of our algorithms. Peršić et al. [2019] use the constant radar-cross section of the tetrahedral target to fit a model that has a quadratic dependency on the received EM wave power (i.e., they model the radar lobe at various distances). To remove the specialized target requirement in this method, the algorithm may need to learn the distribution of the radar lobes and the orientation dependency of the radar cross section for arbitrary objects in the environment by leveraging neural radiance fields, such as Huang et al. [2024]. Using differentiable convex solvers, works, such as Peretroukhin et al. [2020], estimate rotation matrices by learning the cost matrix. The cost matrices of our HE and HERW calibration algorithms are similar to the cost matrix in Peretroukhin et al. [2020], so we could learn them from the raw data.

Appendices

Appendix A

On the Nonlinear Observability of Radar-to-Camera Extrinsic Calibration

In this appendix, we expand the observability analysis from Section 5.4. Following the procedure outlined in Section 4.2.2, we define the system equations, compute the respective Lie derivatives, and demonstrate that the nonlinear observability matrix has full column rank. In the analysis here, the pose, velocity, and acceleration states of the radar-camera system are camera-centric (i.e., taken with respect to the camera and not the radar). Since the camera-centric states can be used to determine the radar-centric states, this change does not affect the observability result.

A.1 Nonlinear Observability Analysis

Given the camera frame $\underrightarrow{\mathcal{F}}_c$, the world frame $\underrightarrow{\mathcal{F}}_w$, and the radar frame $\underrightarrow{\mathcal{F}}_r$, the state vector for the observability analysis is defined as

$$\mathbf{x} = \begin{bmatrix} \mathbf{r}_w^{cw\intercal} & \mathbf{q}_{wc}{}^{\intercal} & \mathbf{v}_w^{cw\intercal} & \boldsymbol{\omega}_c^{cw\intercal} & \mathbf{a}_w^{cw\intercal} & \boldsymbol{\alpha}_c^{cw\intercal} & \gamma & \mathbf{r}_c^{rc\intercal} & \mathbf{q}_{cr}{}^{\intercal} \end{bmatrix}, \tag{A.1}$$

where $\mathbf{r}$, $\mathbf{v}$, and $\mathbf{a}$ denote the translation, linear velocity, and linear acceleration. The vectors $\boldsymbol{\omega}$ and $\boldsymbol{\alpha}$ are the rotational velocity and the rotational acceleration. Finally, γ is the scale factor for the camera translation (for a monocular camera system). The motion model for the system is

$$\dot{\mathbf{x}} = \mathbf{f}_0(\mathbf{x}) + \mathbf{f}_1(\mathbf{x}) = \begin{bmatrix} \mathbf{0}_{3\times1} \\ \frac{1}{2}\Xi(\mathbf{q}_{wc})\boldsymbol{\omega}_c^{cw} \\ \mathbf{0}_{3\times1} \\ \boldsymbol{\alpha}_c^{cw} \\ \mathbf{0}_{3\times1} \\ \mathbf{0}_{3\times1} \\ 0 \\ \mathbf{0}_{3\times1} \\ \mathbf{0}_{4\times1} \end{bmatrix} + \begin{bmatrix} \mathbf{v}_w^{cw} \\ \mathbf{0}_{4\times1} \\ \mathbf{a}_w^{cw} \\ \mathbf{0}_{3\times1} \\ \mathbf{0}_{3\times1} \\ \mathbf{0}_{3\times1} \\ 0 \\ \mathbf{0}_{3\times1} \\ \mathbf{0}_{4\times1} \end{bmatrix}. \tag{A.2}$$

The measurement model equations for the (scaled) camera translation and rotation are, respectively,

$$\begin{aligned}\mathbf{h}_1 &= \gamma\, \mathbf{r}_w^{cw}, \\ \mathbf{h}_2 &= \mathbf{q}_{wc}.\end{aligned} \tag{A.3}$$

Using the camera-centric model, it is possible to directly measure $\mathbf{q}_{wc}$ and determine $\boldsymbol{\omega}_c^{cw}$ and $\boldsymbol{\alpha}_c^{cw}$. Finally, the radar ego-velocity measurement equation is

$$\mathbf{h}_3 = \mathbf{R}^\intercal(\mathbf{q}_{cr})(\mathbf{R}^\intercal(\mathbf{q}_{wc})\mathbf{v}_w^{cw} + \boldsymbol{\omega}_c^{cw\wedge}\mathbf{r}_c^{rc}). \tag{A.4}$$

The observability analysis requires the zeroth, first, and second order Lie derivatives. The zeroth order Lie derivatives are

$$\begin{aligned}\nabla L^0\mathbf{h}_1 &= \begin{bmatrix}\gamma\mathbf{I}_3 & \mathbf{0}_{3\times16} & \mathbf{r}_w^{cw} & \mathbf{0}_{3\times7}\end{bmatrix}, \\ \nabla L^0\mathbf{h}_2 &= \begin{bmatrix}\mathbf{0}_{4\times3} & \mathbf{I}_4 & \mathbf{0}_{4\times20}\end{bmatrix}, \\ \nabla L^0\mathbf{h}_3 &= [\mathbf{0}_{3\times3} \quad \mathbf{A} \quad \mathbf{R}^\intercal(\mathbf{q}_{cr})\mathbf{N} \quad -\mathbf{R}^\intercal(\mathbf{q}_{cr})\mathbf{r}_c^{rc\wedge} \quad \mathbf{0}_{3\times7} \quad \mathbf{R}^\intercal(\mathbf{q}_{cr})\boldsymbol{\omega}_c^{cw\wedge} \quad \mathbf{B}],\end{aligned} \tag{A.5}$$

where

$$\begin{aligned}\mathbf{A} &= \mathbf{R}^\intercal(\mathbf{q}_{cr})\frac{\partial\mathbf{R}^\intercal(\mathbf{q}_{wc})\mathbf{v}_w^{cw}}{\partial\mathbf{q}_{wc}}, \\ \mathbf{B} &= \frac{\partial\mathbf{R}^\intercal(\mathbf{q}_{cr})(\mathbf{R}^\intercal(\mathbf{q}_{wc})\mathbf{v}_w^{cw} + \boldsymbol{\omega}_c^{cw\wedge}\mathbf{r}_c^{rc})}{\partial\mathbf{q}_{cr}}, \\ \mathbf{N} &= \mathbf{R}^\intercal(\mathbf{q}_{wc}).\end{aligned} \tag{A.6}$$

The first order Lie derivatives are

$$\begin{aligned}\nabla L^1_{\mathbf{f}_1}\mathbf{h}_1 &= \begin{bmatrix}\mathbf{0}_{3\times7} & \gamma\mathbf{I}_3 & \mathbf{0}_{3\times9} & \mathbf{v}_w^{cw} & \mathbf{0}_{3\times7}\end{bmatrix}, \\ \nabla L^1_{\mathbf{f}_0}\mathbf{h}_2 &= [\mathbf{0}_{4\times3} \quad \tfrac{1}{2}\Omega(\boldsymbol{\omega}_c^{cw}) \quad \mathbf{0}_{4\times3} \quad \tfrac{1}{2}\Xi(\mathbf{q}_{wc}) \quad \mathbf{0}_{4\times14}], \\ \nabla L^1_{\mathbf{f}_0}\mathbf{h}_3 &= [\mathbf{0}_{3\times3} \quad \mathbf{C} \quad \mathbf{R}^\intercal(\mathbf{q}_{cr})\mathbf{D} \quad \mathbf{E} \quad \mathbf{0}_{3\times3} \quad \mathbf{F} \quad \mathbf{0}_{3\times1} \quad \mathbf{R}^\intercal(\mathbf{q}_{cr})\boldsymbol{\alpha}_c^{cw\wedge} \quad \mathbf{G}], \\ \nabla L^1_{\mathbf{f}_1}\mathbf{h}_3 &= [\mathbf{0}_{3\times3} \quad \mathbf{H} \quad \mathbf{0}_{3\times6} \quad \mathbf{R}^\intercal(\mathbf{q}_{cr})\mathbf{N} \quad \mathbf{0}_{3\times7} \quad \mathbf{L}],\end{aligned} \tag{A.7}$$

where

$$\begin{aligned}\mathbf{D} =& 2(2\dot{\mathbf{q}}_{wc}^0\mathbf{q}_{wc}^0\mathbf{I}_3 + \dot{\mathbf{q}}_{wc}^v\mathbf{q}_{wc}^{v\,\intercal} + \mathbf{q}_{wc}^v\dot{\mathbf{q}}_{wc}^{v\,\intercal} - (\mathbf{q}_{wc}^0\dot{\mathbf{q}}_{wc}^v + \dot{\mathbf{q}}_{wc}^0\mathbf{q}_{wc}^v)^\wedge), \\ \mathbf{G} =& \frac{\partial\mathbf{R}^\intercal(\mathbf{q}_{cr})(\frac{1}{2}\frac{\partial\mathbf{R}^\intercal(\mathbf{q}_{wc})(\mathbf{v}_w^{cw})}{\partial\mathbf{q}_{wc}}\Omega(\boldsymbol{\omega}_c^{cw})\mathbf{q}_{wc} - \mathbf{r}_c^{rc\wedge}\boldsymbol{\alpha}_c^{cw})}{\partial\mathbf{q}_{cr}}, \\ \mathbf{H} =& \mathbf{R}^\intercal(\mathbf{q}_{cr})\frac{\partial\mathbf{R}^\intercal(\mathbf{q}_{wc})\mathbf{a}_w^{cw}}{\partial\mathbf{q}_{wc}}, \\ \mathbf{L} =& \frac{\partial\mathbf{R}^\intercal(\mathbf{q}_{cr})\mathbf{R}^\intercal(\mathbf{q}_{wc})\mathbf{a}_w^{cw}}{\partial\mathbf{q}_{cr}}.\end{aligned} \tag{A.8}$$

We do not explicitly require the non-zero matrices, $\mathbf{C}$, $\mathbf{E}$, and $\mathbf{F}$, in Equation (5.49) because the submatrix formed from the columns corresponding to the rotation states can be shown to be full rank. The second order

Lie derivatives are

$$\begin{aligned}\nabla L^2_{\mathbf{f}_1}\mathbf{h}_1 &= \begin{bmatrix}\mathbf{0}_{3\times 13} & \gamma\mathbf{I}_3 & \mathbf{0}_{3\times 3} & \mathbf{a}_w^{cw} & \mathbf{0}_{3\times 7}\end{bmatrix},\\ \nabla L^2_{\mathbf{f}_0}\mathbf{h}_2 &= [\mathbf{0}_{4\times 3} \quad \mathbf{M} \quad \mathbf{0}_{4\times 3} \quad -\tfrac{1}{2}\mathbf{q}_{wc}\boldsymbol{\omega}_c^{cw\intercal} \quad \mathbf{0}_{4\times 3} \quad \tfrac{1}{2}\Xi(\mathbf{q}_{wc}) \quad \mathbf{0}_{4\times 8}],\end{aligned} \tag{A.9}$$

where

$$\mathbf{M} = \frac{1}{4}(2\boldsymbol{\Omega}(\boldsymbol{\alpha}_c^{cw}) - \boldsymbol{\omega}_c^{cw\intercal}\boldsymbol{\omega}_c^{cw}\mathbf{I}_4). \tag{A.10}$$

Stacking the gradients of the Lie derivatives, we arrive at the nonlinear observability matrix,

$$\mathbf{O} = \begin{bmatrix}\nabla L^0\mathbf{h}_2\\ \nabla L^1_{f_0}\mathbf{h}_2\\ \nabla L^2_{f_0 f_0}\mathbf{h}_2\\ \nabla L^0\mathbf{h}_1\\ \nabla L^1_{f_1}\mathbf{h}_1\\ \nabla L^2_{f_1 f_1}\mathbf{h}_1\\ \nabla L^0\mathbf{h}_3\\ \nabla L^1_{f_0}\mathbf{h}_3\\ \nabla L^1_{f_1}\mathbf{h}_3\end{bmatrix}. \tag{A.11}$$

This matrix is full column rank and hence the system is locally weakly observable. However, we can perform the first few steps of Gaussian row reduction to assist Mathematica. Starting with our observability matrix

$$\mathbf{O} = \begin{bmatrix}
\mathbf{0}_{4\times 3} & \mathbf{I}_4 & \mathbf{0}_{4\times 3} & \mathbf{0}_{4\times 3} & \mathbf{0}_{4\times 3} & \mathbf{0}_{4\times 3} & \mathbf{0}_{4\times 1} & \mathbf{0}_{4\times 3} & \mathbf{0}_{4\times 4}\\
\mathbf{0}_{4\times 3} & \tfrac{1}{2}\Omega(\boldsymbol{\omega}_c^{cw}) & \mathbf{0}_{4\times 3} & \tfrac{1}{2}\Xi(\mathbf{q}_{wc}) & \mathbf{0}_{4\times 3} & \mathbf{0}_{4\times 3} & \mathbf{0}_{4\times 1} & \mathbf{0}_{4\times 3} & \mathbf{0}_{4\times 4}\\
\mathbf{0}_{4\times 3} & \mathbf{M} & \mathbf{0}_{4\times 3} & -\tfrac{1}{2}\mathbf{q}_{wc}\boldsymbol{\omega}_c^{cw\intercal} & \mathbf{0}_{4\times 3} & \tfrac{1}{2}\Xi(\mathbf{q}_{wc}) & \mathbf{0}_{4\times 1} & \mathbf{0}_{4\times 3} & \mathbf{0}_{4\times 4}\\
\gamma\mathbf{I}_3 & \mathbf{0}_{3\times 4} & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} & \mathbf{r}_w^{cw} & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 4}\\
\mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 4} & \gamma\mathbf{I}_3 & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} & \mathbf{v}_w^{cw} & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 4}\\
\mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 4} & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} & \gamma\mathbf{I}_3 & \mathbf{0}_{3\times 3} & \mathbf{a}_w^{cw} & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 4}\\
\mathbf{0}_{3\times 3} & \mathbf{A} & \mathbf{R}^\intercal(\mathbf{q}_{cr})\mathbf{N} & -\mathbf{R}^\intercal(\mathbf{q}_{cr})\mathbf{r}_c^{rc\wedge} & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 1} & \mathbf{R}^\intercal(\mathbf{q}_{cr})\boldsymbol{\omega}_c^{cw\wedge} & \mathbf{B}\\
\mathbf{0}_{3\times 3} & \mathbf{C} & \mathbf{R}^\intercal(\mathbf{q}_{cr})\mathbf{D} & \mathbf{E} & \mathbf{0}_{3\times 3} & \mathbf{F} & \mathbf{0}_{3\times 1} & \mathbf{R}^\intercal(\mathbf{q}_{cr})\boldsymbol{\alpha}_c^{cw\wedge} & \mathbf{G}\\
\mathbf{0}_{3\times 3} & \mathbf{H} & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} & \mathbf{R}^\intercal(\mathbf{q}_{cr})\mathbf{N} & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 1} & \mathbf{0}_{3\times 3} & \mathbf{L}
\end{bmatrix}. \tag{A.12}$$

First, we peform Gaussian row reduction of the rotation submatrix, which results in

$$\mathbf{O} = \begin{bmatrix} \mathbf{0}_{4\times3} & \mathbf{I}_4 & \mathbf{0}_{4\times3} & \mathbf{0}_{4\times3} & \mathbf{0}_{4\times3} & \mathbf{0}_{4\times3} & \mathbf{0}_{4\times1} & \mathbf{0}_{4\times3} & \mathbf{0}_{4\times4} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times4} & \mathbf{0}_{3\times3} & \mathbf{I}_3 & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times1} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times4} \\ \mathbf{0}_{1\times3} & \mathbf{0}_{1\times4} & \mathbf{0}_{1\times3} & \mathbf{0}_{1\times3} & \mathbf{0}_{1\times3} & \mathbf{0}_{1\times3} & 0 & \mathbf{0}_{1\times3} & \mathbf{0}_{1\times4} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times4} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{I}_3 & \mathbf{0}_{3\times1} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times4} \\ \mathbf{0}_{1\times3} & \mathbf{0}_{1\times4} & \mathbf{0}_{1\times3} & \mathbf{0}_{1\times3} & \mathbf{0}_{1\times3} & \mathbf{0}_{1\times3} & 0 & \mathbf{0}_{1\times3} & \mathbf{0}_{1\times4} \\ \gamma\mathbf{I}_3 & \mathbf{0}_{3\times4} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{r}_w^{cw} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times4} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times4} & \gamma\mathbf{I}_3 & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{v}_w^{cw} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times4} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times4} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \gamma\mathbf{I}_3 & \mathbf{0}_{3\times3} & \mathbf{a}_w^{cw} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times4} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{4\times4} & \mathbf{R}^\intercal(\mathbf{q}_{cr})\mathbf{N} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times1} & \mathbf{R}^\intercal(\mathbf{q}_{cr})\boldsymbol{\omega}_c^{cw\wedge} & \mathbf{B} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{4\times4} & \mathbf{R}^\intercal(\mathbf{q}_{cr})\mathbf{D} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times1} & \mathbf{R}^\intercal(\mathbf{q}_{cr})\boldsymbol{\alpha}_c^{cw\wedge} & \mathbf{G} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{4\times4} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{R}^\intercal(\mathbf{q}_{cr})\mathbf{N} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times1} & \mathbf{0}_{3\times3} & \mathbf{L} \end{bmatrix}. \tag{A.13}$$

Second, we reduce the unscaled camera tranlsation terms, so

$$\mathbf{O} = \begin{bmatrix} \mathbf{0}_{4\times3} & \mathbf{I}_4 & \mathbf{0}_{4\times3} & \mathbf{0}_{4\times3} & \mathbf{0}_{4\times3} & \mathbf{0}_{4\times3} & \mathbf{0}_{4\times1} & \mathbf{0}_{4\times3} & \mathbf{0}_{4\times4} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times4} & \mathbf{0}_{3\times3} & \mathbf{I}_3 & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times1} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times4} \\ \mathbf{0}_{1\times3} & \mathbf{0}_{1\times4} & \mathbf{0}_{1\times3} & \mathbf{0}_{1\times3} & \mathbf{0}_{1\times3} & \mathbf{0}_{1\times3} & 0 & \mathbf{0}_{1\times3} & \mathbf{0}_{1\times4} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times4} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{I}_3 & \mathbf{0}_{3\times1} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times4} \\ \mathbf{0}_{1\times3} & \mathbf{0}_{1\times4} & \mathbf{0}_{1\times3} & \mathbf{0}_{1\times3} & \mathbf{0}_{1\times3} & \mathbf{0}_{1\times3} & 0 & \mathbf{0}_{1\times3} & \mathbf{0}_{1\times4} \\ \mathbf{I}_3 & \mathbf{0}_{3\times4} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \frac{1}{\gamma}\mathbf{r}_w^{cw} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times4} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times4} & \mathbf{I}_3 & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \frac{1}{\gamma}\mathbf{v}_w^{cw} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times4} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times4} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{I}_3 & \mathbf{0}_{3\times3} & \frac{1}{\gamma}\mathbf{a}_w^{cw} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times4} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{4\times4} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & -\frac{1}{\gamma}\mathbf{R}^\intercal(\mathbf{q}_{cr})\mathbf{N}\mathbf{v}_w^{cw} & \mathbf{R}^\intercal(\mathbf{q}_{cr})\boldsymbol{\omega}_c^{cw\wedge} & \mathbf{B} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{4\times4} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & -\frac{1}{\gamma}\mathbf{R}^\intercal(\mathbf{q}_{cr})\mathbf{D}\mathbf{v}_w^{cw} & \mathbf{R}^\intercal(\mathbf{q}_{cr})\boldsymbol{\alpha}_c^{cw\wedge} & \mathbf{G} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{4\times4} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & -\frac{1}{\gamma}\mathbf{R}^\intercal(\mathbf{q}_{cr})\mathbf{N}\mathbf{a}_w^{cw} & \mathbf{0}_{3\times3} & \mathbf{L} \end{bmatrix}. \tag{A.14}$$

All that remains is to prove the rank of the bottom right 9 by 8 submatrix

$$O_{\text{sub}} = \begin{bmatrix} -\frac{1}{\gamma}\mathbf{R}^\intercal(\mathbf{q}_{cr})\mathbf{N}\mathbf{v}_w^{cw} & \mathbf{R}^\intercal(\mathbf{q}_{cr})\boldsymbol{\omega}_c^{cw\wedge} & \mathbf{B} \\ -\frac{1}{\gamma}\mathbf{R}^\intercal(\mathbf{q}_{cr})\mathbf{D}\mathbf{v}_w^{cw} & \mathbf{R}^\intercal(\mathbf{q}_{cr})\boldsymbol{\alpha}_c^{cw\wedge} & \mathbf{G} \\ -\frac{1}{\gamma}\mathbf{R}^\intercal(\mathbf{q}_{cr})\mathbf{N}\mathbf{a}_w^{cw} & \mathbf{0}_{3\times3} & \mathbf{L} \end{bmatrix}, \tag{A.15}$$

Using Gaussian row reduction from a symbolic linear algebra software shows that this submatrix is full column rank. Subsequently, the radar-camera extrinsic calibration problem is locally weakly observable. Interestingly, we can clearly see that the system becomes degenerate if the rotational acceleration is coincident with the rotational velocity.

Appendix B

A Local Solver for the HERW Problem

In this appendix, we derive a local solver for the HERW and unscaled HERW problems. For brevity, we derive the optimization problem for the many $\mathbf{X}$s and $\mathbf{Y}$s case. Given a bipartite graph with nodes $\{\mathbf{X}_1, \ldots, \mathbf{X}_{N_i}, \mathbf{Y}_1, \ldots, \mathbf{Y}_{N_j}\}$, let $\mathcal{D}_{i,j}$ be the data that forms the connection between $\mathbf{X}_i$ and $\mathbf{Y}_j$, specifically $\mathcal{D}_{i,j} = \{(\mathbf{A}_{i,j,k}, \mathbf{B}_{i,j,k}) \ \forall \ k = 1, \ldots, N_{i,j}\}$. The noisy measurement models for a given $(\mathbf{A}_{i,j,k}, \mathbf{B}_{i,j,k})$ pair are

$$\begin{aligned}
\mathbf{R}_{\mathbf{B}_{i,j,k}} =& \mathbf{R}_{\mathbf{Y}_j}{}^{\intercal} \mathbf{R}_{\mathbf{A}_{i,j,k}} \mathbf{R}_{\mathbf{X}_i} \exp\left(\mathbf{n}_R^{\wedge}\right), && \text{(B.1)}\\
\mathbf{n}_R \sim& \mathcal{N}\left(0, {\sigma_R}^2 \mathbf{I}\right), && \text{(B.2)}\\
\mathbf{t}_{\mathbf{B}_{i,j,k}} =& \mathbf{R}_{\mathbf{Y}_j}{}^{\intercal} \left(\mathbf{R}_{\mathbf{A}_{i,j,k}} \mathbf{t}_{\mathbf{X}_i} + \mathbf{t}_{\mathbf{A}_{i,j,k}} - \mathbf{t}_{\mathbf{Y}_j}\right) + \mathbf{n}_t, && \text{(B.3)}\\
\mathbf{n}_R \sim& \mathcal{N}\left(0, {\sigma_t}^2 \mathbf{I}\right). && \text{(B.4)}
\end{aligned}$$

Unlike other rotation measurement models in this thesis, this model leverages a right-perturbation noise framework to ensure that the model is an on-manifold analogue of the convex formulation. The associated error distributions are

$$\begin{aligned}
\mathbf{e}_{R_{i,j,k}} =& \log\left(\mathbf{R}_{\mathbf{X}_i}{}^{\intercal} \mathbf{R}_{\mathbf{A}_{i,j,k}}{}^{\intercal} \mathbf{R}_{\mathbf{Y}_j} \mathbf{R}_{\mathbf{B}_{i,j,k}}\right)^{\vee} \sim \mathcal{N}\left(0, {\sigma_R}^2 \mathbf{I}\right), && \text{(B.5)}\\
\mathbf{e}_{t_{i,j,k}} =& \mathbf{R}_{\mathbf{A}_{i,j,k}} \mathbf{t}_{\mathbf{X}_i} + \mathbf{t}_{\mathbf{A}_{i,j,k}} - \mathbf{R}_{\mathbf{Y}_j} \mathbf{t}_{\mathbf{B}_{i,j,k}} - \mathbf{t}_{\mathbf{Y}_j} \sim \mathcal{N}\left(0, {\sigma_t}^2 \mathbf{I}\right). && \text{(B.6)}
\end{aligned}$$

The associated error Jacobians are

$$\begin{aligned}
\frac{\partial \mathbf{e}_{R_{i,j,k}}}{\partial \boldsymbol{\psi}_{\mathbf{X}_i}} =& -\mathbf{R}_{\mathbf{B}_{i,j,k}}{}^{\intercal} \mathbf{R}_{\mathbf{Y}_j}{}^{\intercal} \mathbf{R}_{\mathbf{A}_{i,j,k}}, && \text{(B.7)}\\
\frac{\partial \mathbf{e}_{R_{i,j,k}}}{\partial \boldsymbol{\psi}_{\mathbf{Y}_j}} =& \mathbf{R}_{\mathbf{B}_{i,j,k}}{}^{\intercal} \mathbf{R}_{\mathbf{Y}_j}{}^{\intercal}, && \text{(B.8)}\\
\frac{\partial \mathbf{e}_{t_{i,j,k}}}{\partial \mathbf{t}_{\mathbf{X}_i}} =& \mathbf{R}_{\mathbf{A}_{i,j,k}}, && \text{(B.9)}\\
\frac{\partial \mathbf{e}_{t_{i,j,k}}}{\partial \boldsymbol{\psi}_{\mathbf{Y}_j}} =& \left(\mathbf{R}_{\mathbf{Y}_j} \mathbf{t}_{\mathbf{B}_{i,j,k}}\right)^{\wedge}, && \text{(B.10)}\\
\frac{\partial \mathbf{e}_{t_{i,j,k}}}{\partial \mathbf{t}_{\mathbf{Y}_j}} =& -\mathbf{I}. && \text{(B.11)}
\end{aligned}$$

If the measurement $\mathbf{B}_{i,j,k}$ is from an unscaled pose sensor, then the translation measurement model and

error distribution are

$$\mathbf{t}_{\mathbf{B}_{i,j,k}} = \mathbf{R}_{\mathbf{Y}_j}{}^{\intercal}\left(\mathbf{R}_{\mathbf{A}_{i,j,k}}\mathbf{t}_{\mathbf{X}_i,\alpha} + \alpha\mathbf{t}_{\mathbf{A}_{i,j,k}} - \mathbf{t}_{\mathbf{Y}_j,\alpha}\right) + \mathbf{n}_t, \tag{B.12}$$

$$\mathbf{n}_R \sim \mathcal{N}\left(0, {\sigma_t}^2\mathbf{I}\right), \tag{B.13}$$

$$\mathbf{e}_{t_{i,j,k,\alpha}} = \mathbf{R}_{\mathbf{A}_{i,j,k}}\mathbf{t}_{\mathbf{X}_i,\alpha} + \alpha\mathbf{t}_{\mathbf{A}_{i,j,k}} - \mathbf{R}_{\mathbf{Y}_j}\mathbf{t}_{\mathbf{B}_{i,j,k}} - \mathbf{t}_{\mathbf{Y}_j,\alpha} \sim \mathcal{N}\left(0, {\sigma_t}^2\mathbf{I}\right). \tag{B.14}$$

The associated error Jacobians are

$$\frac{\partial \mathbf{e}_{t_{i,j,k,\alpha}}}{\partial \mathbf{t}_{\mathbf{X}_i,\alpha}} = \mathbf{R}_{\mathbf{A}_{i,j,k}}, \tag{B.15}$$

$$\frac{\partial \mathbf{e}_{t_{i,j,k,\alpha}}}{\partial \boldsymbol{\psi}_{\mathbf{Y}_j}} = \left(\mathbf{R}_{\mathbf{Y}_j}\mathbf{t}_{\mathbf{B}_{i,j,k}}\right)^{\wedge}, \tag{B.16}$$

$$\frac{\partial \mathbf{e}_{t_{i,j,k,\alpha}}}{\partial \mathbf{t}_{\mathbf{Y}_j,\alpha}} = -\mathbf{I}, \tag{B.17}$$

$$\frac{\partial \mathbf{e}_{t_{i,j,k,\alpha}}}{\partial \alpha} = \mathbf{t}_{\mathbf{A}_{i,j,k}}. \tag{B.18}$$

Consequently, the optimization problem is

Problem 23. *Local HERW Optimization Problem*

$\min_{\mathbf{X}_1,\ldots,\mathbf{X}_{N_i},\mathbf{Y}_1,\ldots,\mathbf{Y}_{N_j}} \sum_{i=1}^{N_i}\sum_{j=1}^{N_j}\sum_{k=1}^{N_{i,j}} \frac{1}{\sigma_R^2}\mathbf{e}_{R_{i,j,k}}{}^{\intercal}\mathbf{e}_{R_{i,j,k}} + \frac{1}{\sigma_t^2}\mathbf{e}_{t_{i,j,k}}{}^{\intercal}\mathbf{e}_{t_{i,j,k}}$.

The optimization problem for the unscaled case is

Problem 24. *Local Unscaled HERW Optimization Problem*

$\min_{\mathbf{X}_1,\ldots,\mathbf{X}_{N_i},\mathbf{Y}_1,\ldots,\mathbf{Y}_{N_j},\alpha} \sum_{i=1}^{N_i}\sum_{j=1}^{N_j}\sum_{k=1}^{N_{i,j}} \frac{1}{\sigma_R^2}\mathbf{e}_{R_{i,j,k}}{}^{\intercal}\mathbf{e}_{R_{i,j,k}} + \frac{1}{\sigma_t^2}\mathbf{e}_{t_{i,j,k,\alpha}}{}^{\intercal}\mathbf{e}_{t_{i,j,k,\alpha}}$.

We solve these two optimization problems using the Ceres solver.

Bibliography

Sameer Agarwal, Keir Mierle, and The Ceres Solver Team. Ceres Solver, 3 2022. URL https://github.com/ceres-solver/ceres-solver.

Akshay Agrawal, Brandon Amos, Shane Barratt, Stephen Boyd, Steven Diamond, and J. Zico Kolter. Differentiable convex optimization layers. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, Vancouver, Canada, Dec. 8–14 2019. doi: 10.5555/3454287.3455145.

Erling D. Andersen and Knud D. Andersen. The Mosek interior point optimizer for linear programming: An implementation of the homogeneous algorithm. In *High Performance Optimization*, volume 33, pages 197–232. Springer US, Boston, MA, 2000. doi: 10.1007/978-1-4757-3216-0_8.

Nicolas Andreff, Radu Horaud, and Bernard Espiau. Robot hand-eye calibration using structure-from-motion. *The International Journal of Robotics Research*, 20(3):228–248, 2001. doi: 10.1177/02783640122067372.

Milena Anguelova. *Observability and identifiability of nonlinear systems with applications in biology*. Phd thesis, University of Gothenburg, Gothenburg, Sweden, Nov. 2007. http://www.math.chalmers.se/Math/Research/Preprints/Doctoral/2007/3.pdf.

F. Anstett-Collin, L. Denis-Vidal, and G. Millérioux. A priori identifiability: An overview on definitions and approaches. *Annual Reviews in Control*, 50:139–149, 2020. doi: 10.1016/j.arcontrol.2020.10.006.

Timothy D Barfoot. *State estimation for robotics*. Cambridge Univ. Press, Cambridge, UK, 2024. doi: 10.1017/9781316671528.

Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004. doi: 10.1017/CBO9780511804441.

Jesus Briales and Javier Gonzalez-Jimenez. Cartan-sync: Fast and global $\mathrm{SE}(d)$-synchronization. *IEEE Robotics and Automation Letters*, 2(4):2127–2134, 2017a. doi: 10.1109/LRA.2017.2718661.

Jesus Briales and Javier Gonzalez-Jimenez. Convex global 3D registration with Lagrangian duality. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5612–5621, Honolulu, HI, USA, Jul. 21–26 2017b. doi: 10.1109/CVPR.2017.595.

Jesus Briales, Laurent Kneip, and Javier Gonzalez-Jimenez. A certifiably globally optimal solution to the non-minimal relative pose problem. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 145–154, Salt Lake City, UT, USA, Jun. 18–22 2018. doi: 10.1109/CVPR.2018.00023.

Jonathan Brookshire and Seth Teller. Automatic calibration of multiple coplanar sensors. *Robotics: Science and Systems VII*, 33, Jul. 2011. doi: 10.7551/mitpress/9481.003.0010.

Jonathan Brookshire and Seth Teller. Extrinsic calibration from per-sensor egomotion. *Robotics: Science and Systems VIII*, pages 504–512, Jul. 2012. doi: 10.7551/mitpress/9816.003.0009.

Keenan Burnett, David J Yoon, Yuchen Wu, Andrew Z Li, Haowei Zhang, Shichen Lu, Jingxing Qian, Wei-Kang Tseng, Andrew Lambert, Keith YK Leung, Angela P Schoellig, and Timothy D Barfoot. Boreas: A multi-season autonomous driving dataset. *The International Journal of Robotics Research*, 42:33–42, 2023. doi: 10.1177/02783649231160195.

Carlos Campos, Richard Elvira, Juan J. Gómez Rodr'iguez, José M. M. Montiel, and Juan D. Tardós. ORB-SLAM3: An accurate open-source library for visual, visual–inertial, and multimap SLAM. *IEEE Transactions on Robotics*, 37(6):1874–1890, May 2021. doi: 10.1109/TRO.2021.3075644.

Qilong Cheng, Emmett Wise, and Jonathan Kelly. Extrinsic calibration of 2D millimetre-wavelength radar pairs using ego-velocity estimates. In *IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, pages 559–565, Seattle, WA, USA, 2023. doi: 10.1109/AIM46323.2023.10196187.

A. Chiuso, P. Favaro, Hailin Jin, and S. Soatto. Structure from motion causally integrated over time. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4):523–535, Apr. 2002. doi: 10.1109/34.993559.

Diego Cifuentes, Sameer Agarwal, Pablo A. Parrilo, and Rekha R. Thomas. On the local stability of semidefinite relaxations. *Mathematical Programming*, 193(2):629–663, Sep 2021. doi: 10.1007/s10107-021-01696-1.

Marta Čolaković-Bencerić, Juraj Peršić, Ivan Marković, and Ivan Petrović. On hand-eye calibration via on-manifold gauss-newton optimization. In *Intelligent Autonomous Systems 17*, pages 378–391, Zagreb, Croatia, Jun. 13-16 2023. doi: 10.1007/978-3-031-22216-0_26.

Konstantinos Daniilidis. Hand-eye calibration using dual quaternions. *The International Journal of Robotics Research*, 18(3):286–298, Mar. 1999. doi: 10.1177/02783649922066213.

Carl de Boor. *A Practical Guide to Splines*, volume 27. Springer-Verlag, Jan. 1978.

Christopher Doer and Gert F. Trommer. Radar inertial odometry with online calibration. In *European Navigation Conference (ENC)*, pages 1–10, Nov. 23–24 2020a. doi: 10.23919/ENC48637.2020.9317343.

Christopher Doer and Gert F. Trommer. An EKF based approach to radar inertial odometry. In *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, pages 152–159, Karlsruhe, Germany, Sep. 14–16 2020b. doi: 10.1109/MFI49285.2020.9235254.

Christopher Doer and Gert F. Trommer. Radar visual inertial odometry and radar thermal inertial odometry: Robust navigation even in challenging visual conditions. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 331–338, Prague, Czech Republic, Sep. 27 – Oct. 1 2021. doi: 10.1109/IROS51168.2021.9636799.

Kevin J. Doherty, David M. Rosen, and John J. Leonard. Spectral measurement sparsification for pose-graph slam. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 01–08, Kyoto, Japan, Oct. 23–27 2022. doi: 10.1109/IROS47612.2022.9981584.

J. Domhof, J. F. P. Kooij, and D. M. Gavrila. An extrinsic calibration tool for radar, camera and lidar. In *International Conference on Robotics and Automation (ICRA)*, pages 8107–8113, Montréal, Canada, May, 20–24 2019. doi: 10.1109/ICRA.2019.8794186.

F. Dornaika and R. Horaud. Simultaneous robot-world and hand-eye calibration. *IEEE Transactions on Robotics and Automation*, 14(4):617–622, 1998. doi: 10.1109/70.704233.

Ghina El Natour, Omar Ait Aider, Raphael Rouveure, François Berry, and Patrice Faure. Radar and vision sensors calibration for outdoor 3D reconstruction. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 2084–2089, Seattle, WA, USA, May 25–30 2015. doi: 10.1109/ICRA.2015.7139473.

Daniele Evangelista, Emilio Olivastri, Davide Allegro, Emanuele Menegatti, and Alberto Pretto. A graph-based optimization framework for hand-eye calibration for multi-camera setups. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 11474–11480, London, UK, May 29 – Jun. 2 2023. doi: 10.1109/ICRA48891.2023.10160758.

Paul L Fackler. Notes on matrix calculus. *North Carolina State University*, 2005.

Christian Forster, Luca Carlone, Frank Dellaert, and Davide Scaramuzza. On-manifold preintegration for real-time visual–inertial odometry. *IEEE Transactions on Robotics*, 33(1):1–21, 2017. doi: 10.1109/TRO.2016.2597321.

Fadri Furrer, Marius Fehr, Tonci Novkovic, Hannes Sommer, Igor Gilitschenski, and Roland Siegwart. Evaluation of combined time-offset estimation and hand-eye calibration on robotic datasets. In *Field and Service Robotics*, volume 5, pages 145–159. Zurich, Switzerland, Sep. 12–15 2018. doi: 10.1007/978-3-319-67361-5_10.

Michael Garstka, Mark Cannon, and Paul Goulart. COSMO: A conic operator splitting method for convex conic problems. *Journal of Optimization Theory and Applications*, 190(3):779–810, 2021. doi: 10.1007/s10957-021-01896-x.

Matthew Giamou. *Semidefinite Relaxations for Geometric Problems in Robotics*. Phd thesis, University of Toronto, Toronto, Canada, Mar. 2023. http://hdl.handle.net/1807/126896.

Matthew Giamou, Ziye Ma, Valentin Peretroukhin, and Jonathan Kelly. Certifiably globally optimal extrinsic calibration from per-sensor egomotion. *IEEE Robotics and Automation Letters*, 4(2):367–374, 2019. doi: 10.1109/LRA.2018.2890444.

Christopher Grebe. A study of observability-aware trajectory optimization. Master's thesis, University of Toronto, Toronto, Canada, Jan. 2022. http://hdl.handle.net/1807/110811.

Jan Heller, Didier Henrion, and Tomas Pajdla. Hand-eye and robot-world calibration by global polynomial optimization. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 3157–3164, Hong Kong, China, May 31 – Jun 5 2014. doi: 10.1109/ICRA.2014.6907313.

Lionel Heng. Automatic targetless extrinsic calibration of multiple 3D lidars and radars. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10669–10675, Las Vegas, NV, USA, Oct. 25, 2020–Jan. 24, 2021 2020. doi: 10.1109/IROS45743.2020.9340866.

R. Hermann and A. Krener. Nonlinear controllability and observability. *IEEE Transactions on Automatic Control*, 22(5):728–740, Oct. 1977. doi: 10.1109/TAC.1977.1101601.

Robert A. Hewitt and Joshua A. Marshall. Towards intensity-augmented SLAM with LiDAR and ToF sensors. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1957–1961, Hamburg, Germany, Sep. 28 – Oct. 2 2015. doi: 10.1109/IROS.2015.7353634.

Radu Horaud and Fadi Dornaika. Hand-eye calibration. *The International Journal of Robotics Research*, 14(3): 195–210, 1995. doi: 10.1177/027836499501400301.

Berthold KP Horn, Hugh M Hilden, and Shahriar Negahdaripour. Closed-form solution of absolute orientation using orthonormal matrices. *Journal of the Optical Society of America A*, 5(7):1127–1135, Jul 1988. doi: 10.1364/JOSAA.5.001127.

Markus Horn, Thomas Wodtko, Michael Buchholz, and Klaus Dietmayer. Extrinsic infrastructure calibration using the hand-eye robot-world formulation. In *IEEE Intelligent Vehicles Symposium (IV)*, pages 1–8, Anchorage, AK, USA, Jun. 4–7 2023. doi: 10.1109/IV55152.2023.10186703.

Xiao Hu, Daniel Olesen, and Knudsen Per. A novel robust approach for correspondence-free extrinsic calibration. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Macau, China, Nov. 4–8 2019. doi: 10.1109/IROS40897.2019.8968447.

Tianshu Huang, John Miller, Akarsh Prabhakara, Tao Jin, Tarana Laroia, Zico Kolter, and Anthony Rowe. DART: Implicit doppler tomography for radar novel view synthesis. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 24118–24129, Los Alamitos, CA, USA, Jun 2024. doi: 10.1109/CVPR52733.2024.02277.

John A. Jacquez. Identifiability and parameter estimation. *Journal of Parenteral and Enteral Nutrition*, 15(3): 55S–59S, 1991. doi: 10.1177/014860719101500355S.

D. Kellner, M. Barjenbruch, J. Klappstein, J. Dickmann, and K. Dietmayer. Instantaneous ego-motion estimation using doppler radar. In *16th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 869–874, The Hague, Netherlands, Oct. 6–9 2013. doi: 10.1109/ITSC.2013.6728341.

D. Kellner, M. Barjenbruch, K. Dietmayer, J. Klappstein, and J. Dickmann. Joint radar alignment and odometry calibration. In *18th International Conference on Information Fusion (FUSION)*, pages 366–374, Washington, DC, USA, Jul. 6–9 2015.

Jonathan Kelly and Gaurav S Sukhatme. Visual-inertial sensor fusion: Localization, mapping and sensor-to-sensor self-calibration. *The International Journal of Robotics Research*, 30(1):56–79, 2011. doi: 10.1177/0278364910382802.

Jonathan Kelly, Christopher Grebe, and Matthew Giamou. A question of time: Revisiting the use of recursive filtering for temporal calibration of multisensor systems. In *Proceedings of the IEEE International Conference on Multisensor Fusion and Integration (MFI)*, Karlsruhe, Germany, Sep. 23–25 2021. doi: 10.1109/MFI52462.2021.9591176.

Du Yong Kim and Moongu Jeon. Data fusion of radar and image measurements for multi-object tracking via Kalman filtering. *Information Sciences*, 278:641–652, Sep. 2014. doi: 10.1016/j.ins.2014.03.080.

Jihun Kim, Dong Seog Han, and Benaoumeur Senouci. Radar and vision sensor fusion for object detection in autonomous vehicle surroundings. In *4th International Conference on Ubiquitous and Future Networks (ICUFN)*, pages 76–78, Prague, Czech Republic, Jul. 3–6 2018. doi: 10.1109/ICUFN.2018.8436959.

Taehwan Kim, Sungho Kim, Eunryung Lee, and Miryong Park. Comparative analysis of RADAR-IR sensor fusion methods for object detection. In *17th International Conference on Control, Automation and Systems (ICCAS)*, pages 1576–1580, Jeju, Korea, Oct. 18–21 2017. doi: 10.23919/ICCAS.2017.8204237.

Jacob Lambert, Lee Clement, Matthew Giamou, and Jonathan Kelly. Entropy-based $\mathrm{Sim}(3)$ calibration of 2D lidars to egomotion sensors. In *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, pages 455–461, Baden-Baden, Germany, Sep. 19–21 2016. doi: 10.1109/MFI.2016.7849530.

Chia-Le Lee, Yu-Han Hsueh, Chieh-Chih Wang, and Wen-Chieh Lin. Extrinsic and temporal calibration of automotive radar and 3D lidar. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 9976–9983, Las Vegas, NV, USA, Oct. 25, 2020–Jan. 24, 2021 2020. doi: 10.1109/IROS45743.2020.9341715.

Ai Guo Li, Lin Wang, and Defeng Wu. Simultaneous robot-world and hand-eye calibration using dual-quaternions and kronecker product. *International Journal of Physical Sciences*, 5:1530–1536, 2010. doi: 10.5897/IJPS.9000501.

Mingyang Li and Anastasios I. Mourikis. Online temporal calibration for camera-IMU systems: Theory and algorithms. *International Journal of Robotics Research*, 33(7):947–964, 2014. doi: 10.1177/0278364913515286.

Jordan Marr and Jonathan Kelly. Unified spatiotemporal calibration of monocular cameras and planar lidars. In *Proceedings of the International Symposium on Experimental Robotics*, volume 11, pages 781–790. Springer, Cham, Nov. 5–8 2018. doi: 10.1007/978-3-030-33950-0_67.

Huy Nguyen and Quang-Cuong Pham. On the covariance of $\mathbf{X}$ in $\mathbf{AX} = \mathbf{XB}$. *IEEE Transactions on Robotics*, 34(6):1651–1658, 2018. doi: 10.1109/TRO.2018.2861905.

J. Oh, K. Kim, M. Park, and S. Kim. A comparative study on camera-radar calibration methods. In *15th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, pages 1057–1062, Singapore, Nov. 18–21 2018. doi: 10.1109/ICARCV.2018.8581329.

Edwin Olson. AprilTag: A robust and flexible visual fiducial system. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 3400–3407, Shanghai, China, May 9–13 2011. doi: 10.1109/ICRA.2011.5979561.

Kunle T. Olutomilayo, Mojtaba Bahramgiri, Saeid Nooshabadi, and Daniel R. Fuhrmann. Extrinsic calibration of radar mount position and orientation with multiple target configurations. *IEEE Transactions on Instrumentation and Measurement*, 70:1–13, 2021. doi: 10.1109/TIM.2021.3077657.

Luc Oth, Paul Furgale, Laurent Kneip, and Roland Siegwart. Rolling shutter camera calibration. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1360–1367, Portland, Oregon, Jun. 23–28 2013. doi: 10.1109/CVPR.2013.179.

F.C. Park and B.J. Martin. Robot sensor calibration: solving AX=XB on the Euclidean group. *IEEE Transactions on Robotics and Automation*, 10(5):717–721, 1994. doi: 10.1109/70.326576.

Valentin Peretroukhin, Matthew Giamou, W. Nicholas Greene, David Rosen, Jonathan Kelly, and Nicholas Roy. A smooth representation of belief over so(3) for deep rotation learning with uncertainty. In *Robotics: Science and Systems XVI*, RSS2020. Robotics: Science and Systems Foundation, Jul. 2020. doi: 10.15607/rss.2020.xvi.007.

Juraj Peršić, Ivan Marković, and Ivan Petrović. Extrinsic 6DoF calibration of a radar–lidar–camera system enhanced by radar cross section estimates evaluation. *Robotics and Autonomous Systems*, 114:217–230, Apr. 2019. doi: 10.1016/j.robot.2018.11.023.

Juraj Peršić, L. Petrović, I. Marković, and I. Petrović. Online multi-sensor calibration based on moving object tracking. *Advanced Robotics*, 35(3-4):130–140, 2021a. doi: 10.1080/01691864.2020.1819874.

Juraj Peršić, Luka Petrović, Ivan Marković, and Ivan Petrović. Spatiotemporal multisensor calibration via Gaussian processes moving target tracking. *IEEE Transactions on Robotics*, 37(5):1401–1415, Mar. 2021b. doi: 10.1109/TRO.2021.3061364.

Nikita Petrov, Oleg Krasnov, and Alexander G. Yarovoy. Auto-calibration of automotive radars in operational mode using simultaneous localisation and mapping. *IEEE Transactions on Vehicular Technology*, 70(3):2062–2075, 2021. doi: 10.1109/TVT.2021.3058778.

Luis Pineda, Taosha Fan, Maurizio Monge, Shobha Venkataraman, Paloma Sodhi, Ricky T. Q. Chen, Joseph Ortiz, Daniel DeTone, Austin Wang, Stuart Anderson, Jing Dong, Brandon Amos, and Mustafa Mukadam. Theseus: a library for differentiable nonlinear optimization. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, New Orleans, LA, USA, Nov. 28 – Dec. 9 2022. doi: 10.5555/3600270.3600545.

Kaihuai Qin. General matrix representations for b-splines. In *6th Pacific Conference on Computer Graphics and Applications*, pages 37–43, Singapore, Oct. 26–29 1998. doi: 10.1109/PCCGA.1998.731996.

Robert M. Rauber and Stephen W. Nesbitt. *Radar Meteorology: A First Course*. John Wiley & Sons, Ltd, Chichester, United Kingdom, 2018. doi: 10.1002/9781118432662.ch6.

Joern Rehder, Janosch Nikolic, Thomas Schneider, Timo Hinzmann, and Roland Siegwart. Extending Kalibr: Calibrating the extrinsics of multiple IMUs and of individual axes. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 4304–4311, Stockholm, Sweden, May 16–20 2016a. doi: 10.1109/ICRA.2016.7487628.

Joern Rehder, Roland Siegwart, and Paul Furgale. A general approach to spatiotemporal calibration in multisensor systems. *IEEE Transactions on Robotics*, 32(2):383–398, 2016b. doi: 10.1109/TRO.2016.2529645.

Mark A. Richards, James A. Scheer, and William A. Holm, editors. *Basic principles*, volume 1 of *Principles of Modern Radar*. Institution of Eng. and Technol., 2010.

David M Rosen, Luca Carlone, Afonso S Bandeira, and John J Leonard. SE-Sync: A certifiably correct algorithm for synchronization over the special Euclidean group. *International Journal of Robotics Research*, 38(2-3):95–125, March 2019. doi: 10.1177/0278364918784361.

C. Schöller, M. Schnettler, A. Krämmer, G. Hinz, M. Bakovic, M. Güzet, and A. Knoll. Targetless rotational auto-calibration of radar and camera for intelligent transportation systems. In *IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 3934–3941, Auckland, New Zealand, Oct. 27–30 2019. doi: 10.1109/ITSC.2019.8917135.

Mili Shah. Solving the Robot-World/Hand-Eye Calibration Problem Using the Kronecker Product. *Journal of Mechanisms and Robotics*, 5(3):031007, 06 2013. doi: 10.1115/1.4024473.

Joan Solà. Quaternion kinematics for the error-state Kalman filter, Nov. 3 2017. doi: 10.48550/arXiv.1711.02508.

Joan Solà, Jeremie Deray, and Dinesh Atchuthan. A micro Lie theory for state estimation in robotics, Dec. 8 2021. doi: 10.48550/arXiv.1812.01537.

C. Sommer, V. Usenko, D. Schubert, N. Demmel, and D. Cremers. Efficient derivative computation for cumulative B-splines on Lie groups. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11145–11153, Jun. 14–19 2020. doi: 10.1109/CVPR42600.2020.01116.

Carl C. Stahoviak. An instantaneous 3D ego-velocity measurement algorithm for frequency modulated continuous wave (FMCW) doppler radar data. Master's thesis, University of Colorado at Boulder, 2019.

Klaus H. Strobl and Gerd Hirzinger. Optimal hand-eye calibration. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4647–4653, Beijing, China, Oct. 9–15 2006. doi: 10.1109/IROS.2006.282250.

Jos F. Sturm. Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11(1-4):625–653, 1999. doi: 10.1080/10556789908805766.

Shigeki Sugimoto, Hayato Tateda, Hidekazu Takahashi, and Masatoshi Okutomi. Obstacle detection using millimeter-wave radar and its visualization on image sequence. In *International Conference on Pattern Recognition (ICPR)*, pages 342–345, Cambridge, England, Aug. 23–26 2004. doi: 10.1109/ICPR.2004.1334537.

Richard Szeliski. *Computer Vision*. Springer Cham, Switzerland, 2022. doi: 10.1007/978-3-030-34372-9.

Amy Tabb and Khalil M. Ahmad Yousef. Solving the robot-world hand-eye(s) calibration problem with iterative methods. *Machine Vision and Applications*, 28(5–6):569–590, Aug. 2017. doi: 10.1007/s00138-017-0841-7.

Texas Instruments. *xWR1843 Evaluation Module (xWR1843BOOST) Single-Chip mmWave Sensing Solution*, May 2020. URL https://www.ti.com/lit/ug/spruim4b/spruim4b.pdf.

Texas Instruments. *IWR6843AOP Single-Chip 60- to 64-GHz mmWave Sensor Antennas-On-Package (AOP)*, Jul. 2022. URL https://www.ti.com/lit/ds/symlink/iwr6843aop.pdf.

K. C. Toh, M. J. Todd, and R. H. Tütüncü. SDPT3 — A MATLAB software package for semidefinite programming, Version 1.3. *Optimization Methods and Software*, 11(1-4):545–581, Jan. 1999. doi: 10.1080/10556789908805762.

R.Y. Tsai and R.K. Lenz. A new technique for fully autonomous and efficient 3D robotics hand/eye calibration. *IEEE Transactions on Robotics and Automation*, 5(3):345–358, Jun. 1989. doi: 10.1109/70.34770.

Celyn Walters, Oscar Mendez, Simon Hadfield, and Richard Bowden. A robust extrinsic calibration framework for vehicles with unscaled sensors. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 36–42, Macau, China, Nov. 4–8 2019. doi: 10.1109/IROS40897.2019.8968244.

Tao Wang, Nanning Zheng, Jingmin Xin, and Zheng Ma. Integrating millimeter wave radar with a monocular vision sensor for on-road obstacle detection applications. *Sensors*, 11(9):8992–9008, Sep. 2011. doi: 10.3390/s110908992.

Yifu Wang, Wenqing Jiang, Kun Huang, Sören Schwertfeger, and Laurent Kneip. Accurate calibration of multi-perspective cameras from a generalization of the hand-eye constraint. In *International Conference on Robotics and Automation (ICRA)*, pages 1244–1250, Philadelphia, PA, USA, May 23–27 2022. doi: 10.1109/ICRA46639.2022.9811577.

Li Wei, LU Naiguang, Dong Mingli, and Lou Xiaoping. Calibration-free robot-sensor calibration approach based on second-order cone programming. In *MATEC Web of Conferences*, volume 173. EDP Sciences, 2018. doi: 10.1051/matecconf/201817302005.

E. Wise, M. Giamou, S. Khoubyarian, A. Grover, and J. Kelly. Certifiably optimal monocular hand-eye calibration. In *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, pages 271–278, Karlsruhe, Germany, Sep. 14–16 2020. doi: 10.1109/MFI49285.2020.9235219.

Emmett Wise, Juraj Peršić, Christopher Grebe, Ivan Petrović, and Jonathan Kelly. A continuous-time approach for 3D radar-to-camera extrinsic calibration. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 13164–13170, Xi'an, China, May 30–Jun. 5 2021. doi: 10.1109/ICRA48506.2021.9561938.

Emmett Wise, Qilong Cheng, and Jonathan Kelly. Spatiotemporal calibration of 3D millimetre-wavelength radar-camera pairs. *IEEE Transactions on Robotics*, pages 1–15, 2023. doi: 10.1109/TRO.2023.3311680.

Thomas Wodtko, Markus Horn, Michael Buchholz, and Klaus Dietmayer. Globally optimal multi-scale monocular hand-eye calibration using dual quaternions. In *International Conference on 3D Vision (3DV)*, pages 249–257, Dec. 1–3 2021. doi: 10.1109/3DV53792.2021.00035.

Heng Yang, Jingnan Shi, and Luca Carlone. Teaser: Fast and certifiable point cloud registration. *IEEE Transactions on Robotics*, 37(2):314–333, 2021. doi: 10.1109/TRO.2020.3033695.