Daniel Bjerkehagen

Automatic Calibration of Ship-mounted Cameras' Extrinsic Parameters

Master's thesis in Cybernetics and Robotics
Supervisor: Edmund Førland Brekke
Co-supervisor: Esten Ingar Grøtli, Johannes Tjønnås
June 2023

Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Engineering Cybernetics

NTNU
Norwegian University of
Science and Technology

Daniel Bjerkehagen

Automatic Calibration of Ship-mounted Cameras' Extrinsic Parameters

Master's thesis in Cybernetics and Robotics
Supervisor: Edmund Førland Brekke
Co-supervisor: Esten Ingar Grøtli, Johannes Tjønnås
June 2023

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Engineering Cybernetics

NTNU
Norwegian University of
Science and Technology

Abstract

Calibration is a central topic for the field of autonomous systems, as an assumption of accurate calibration is an assumption that is made whenever sensor data is used to build a model of the world around the system. Despite this, calibration is mostly performed by hand and by using specialized methods and infrastructure. This is inefficient and if the properties of the sensor changes, even if only slightly, during operation, then using the now wrongful calibration can have fatal consequences. One set of calibration parameters that all sensors have are the extrinsic calibration parameters, meaning the position and orientation of the sensor.

The Hand-Eye calibration problem is a mathematical equation whose solution is the unknown extrinsic calibration of a sensor. If the sensor to be calibrated is capable of estimating its egomotion, then the equation can be solved entirely by the use of captured data. Such a method for data-driven estimation of the extrinsics enables on-line fault detection and re-calibration, and the Hand-Eye equation thus motivates further study. The extrinsic parameters are however unobservable through Hand-Eye calibration when purely planar motions are concerned, which leads to poor performance for the nearly planar ship-data in focus for this thesis.

Previous work by the author in the specialization project and associated conference paper has shown it possible to perform Hand-Eye calibration to find the orientation of ship-mounted camera, when the camera motion is reconstructed using egomotion algorithms. This thesis entails in-depth analyses of the specific challenges of using the nearly planar ship-data for Hand-Eye calibration. This is done by analysing the Park-Martin Hand-Eye solver in detail. New methods regarding the numerical properties of data for use in Hand-Eye calibration are derived and used to further improve the possibility of on-line, real-time calibration of ship-mounted cameras' extrinsic parameters through purely data-driven, and therefore automatic, methods.

Sammendrag

Norwegian translation of the abstract.
Kalibrering er et sentralt tema innenfor autonome systemer som følge av at antakelsen om kalibrerte sensorer alltid er tilstede når sensordata brukes til å utvikle en modell av verden rundt det autonome systemet. Til tross for dette gjøres kalibrering i dag stort sett for hånd ved hjelp av spesialiserte metoder og infrastruktur. Dette er ineffektivt og dersom sensorens egenskaper endres – selv en liten endring – under operasjonen av det autonome systemet, vil den nå inkorrekte kalibreringen kunne ha fatale konsekvenser. Et sett kalibreringsparametere felles for alle sensorer er de ekstrinsiske kalibreringsparameterne, det vil si posisjonen og orienteringen av sensoren.

Hånd-Øye-kalibreringsproblemet er en matematisk ligning, hvor dens løsning er den ukjente ekstrinsiske kalibreringen av sensoren. Dersom sensoren som kalibreres har evnen til å estimere sin egen bevegelse kan ligningen løses i sin helhet ved hjelp av innsamlet data. En slik metode for datadrevet estimering av de ekstrinsiske parametrene muliggjør feildetektering og re-kalibrering under kjøring av det autonome systemet. Dermed motiverer Hånd-Øye-ligningen til videre undersøkelse. De ekstrinsiske parameterne er dog ikke observerbare for Hånd-Øye-kalibrering når det gjelder rent planare bevegelser, noe som fører til dårlig ytelse for den nært planare skipsdataen som er fokuset i denne oppgaven.

Gjennom prosjektoppgaven og en tilknyttet konferanseartikkel har forfatteren vist at det er mulig å utføre Hånd-Øye-kalibrering for å finne orienteringen av et skipsmontert kamera når kamerabevegelsen er rekonstruert ved å estimere kameraets bevegelse ved hjelp av dens egen data. Denne masteroppgaven tar for seg en dybdeanalyse av utfordringene knyttet til bruken av nært planare skipsdata for Hånd-Øye-kalibrering. Dette er gjennomført ved å analysere Hånd-Øye-løseren kalt Park-Martin i detalj. Gjennom denne analysen utledes nye metoder for å undersøke de numeriske egenskapene til skipsdataen. Disse metodene er deretter brukt til å videre muliggjøre online sanntidskalibrering av skipsmonterte kamera sine ekstrinsiske parametre gjennom rent datadrevne, og dermed autonome, metoder.

Forord

Med denne avhandlingen avsluttes fem år med studier ved Institutt for Teknisk Kybernetikk, NTNU Trondheim. Masteroppgaven bygger på prosjektoppgaven skrevet høsten 2022, som igjen er basert på prosjektet jeg jobbet med sommeren 2022 da jeg hadde sommerjobb i SINTEF Digital og gruppen “Robotics and control”. Av den grunn er motivasjonen for arbeidet, og dermed introduksjonskapitlene i oppgavene, svært like.

Jeg ønsker å utrekke en takk til mine veiledere – Edmund, Esten og Johannes – for gode innspill og godt samarbeid. Spesiell takk rettes også til innsatsen, initiativet, for ikke å nevne midlene(!), lagt ned for å sende meg til FUSION2023-konferansen med en artikkel basert på prosjektoppgaven i hånden. Det hadde aldri skjedd uten dere. En takk går også til Torbjørn Barheim og kolleger ved Kongsberg Maritime avd. Seatex for hjelp med datasett. Dette arbeidet hadde heller ikke vært mulig uten SFI Autoship og det samarbeidet mellom NTNU, SINTEF og Kongsberg som forskningsenteret muliggjør.

Når jeg tenker tilbake på hva det er som har definert studentlivet de siste fem årene så lander jeg alltid på at det er folkene. Studentene har et særegent “få til”-driv som resten av verden bør misunne. Så til alle som bidrar til dette: Takk til dere!

Til slutt utrekkes en endeløs takk til min samboer Sif. Uten din støtte hadde jeg vært halvveis av der jeg er i dag.

Contents

Acronyms

BCH Baker–Campbell–Hausdorf. 11, 38, 39, 82

HE Hand-Eye. 1, 2, 39, 53, 56, 66, 69, 71, 74, 81–87

HT Homogeneous Transformation. 11, 12, 16, 17, 56, 60

KM Kongsberg Maritime. 61, 62

NED North-East-Down. 7, 16, 17, 59, 60, 70

NTNU Norwegian University of Science and Technology. 61

SfM Structure from Motion. 59–63, 82

Chapter 1

Introduction

The field of autonomous systems is rapidly evolving, and in Norway, the focus is in large part on autonomous ships. With Norway's long coast and naval traditions, this is not a surprising development. Today, several projects with semi- or fully autonomous ships are being tested in Norwegian waters, with projects such as Yara Birkeland and Asko Maritime paving the way for efficient and innovative transport of people and goods. These advances have been in part, or even fully, enabled by advances in sensor fusion and as a consequence the accuracy of the calibration of these sensors is more vital to operations than ever.

Any sensor is in principle a method of perceiving the environment given some assumed relationship, the measurement model, between the environment and the obtained measurement data. This way, sensors act as an autonomous system's eyes and ears in enabling perception of the environment, decision-making, and navigation, among others. To do these tasks safely, efficiently and confidently requires accurate calibration of the sensor-suite. Accurate calibration of the sensors ensures their assumed measurement model is as reflective of reality as possible, allowing for safe use of the measurements. All sensors have a specific set of parameters that make up the *calibration* of that sensor, and typically all these parameters must be calibrated in a way unique to the sensor. However, one important set of calibration-parameters that is calibrated more or less the same way for all sensors are the *extrinsic parameters*, meaning the position and orientation of the sensor. To illustrate the importance of accurate knowledge of sensors' extrinsics, consider an algorithm using camera images for automatically detecting obstacles in front of an autonomous car. If the orientation of the camera shifts during operations, the perceived position of obstacles is no longer accurate. This can lead to dangerous and even fatal consequences.

A popular method for performing calibration of the extrinsic parameters of a camera, when said camera is mounted on a robotic arm, involves solving the *Hand-Eye calibration problem*. This problem is a mathematical matrix equation, for which the unknown variable is the extrinsic calibration [1]. The equation is simple and concise, and is in principle only based on a handful of equalities. Research on the problem since its inception in 1989 has mostly focused on specialized solvers in an attempt to estimate the extrinsics with lower error and shorter runtime [2, 3], and some solvers are able to get as close as within 0.1° and 2 mm of the ground truth parameters in optimal, controlled experiments. The problem formulation mostly makes use of the ability to estimate the pose of a camera between two images [4], but in principle any sensor whose data enables *egmotion estimation* can have its extrinsics solved by Hand-Eye calibration.

Egomotion estimation is the problem of finding the motion of a sensor relative an often assumed static environment. For cameras, this can be done efficiently when a calibration plate is available, but multiple algorithms exist for performing egomotion estimation even in unstructured environments. This is often performed by detecting and tracking geometric features in the images, and such methods have enabled the use of the dense camera information.

Previous work by the author has shown how it is possible to recognize that a ship with a mounted camera has the same set of measurements as the classical Hand-Eye problem formulation. In the specialization project leading up to this thesis it was shown how one could go about estimating the camera orientation of a ship-mounted camera by formulating and solving a fitting Hand-Eye calibration problem, a conclusion corroborated by simulations using real-world data resulting in around 2° estimation error. Using the Hand-Eye framework with camera for this setup is advantageous in multiple ways, most chiefly in it being purely data-driven and as such requires no extra infrastructure neither on the ship nor in the dock. But utilizing the HE calibration problem to this end also highlighted some important questions unanswerable by established theory. It has been known since the first papers on the problem that only two motions of the system of non-parallel axes of rotation are enough to make the orientation of the camera relative the robotic arm observable [1, 5], but not much literature exists on quantifying the effect of the *nearly* parallel axes that ship motions result in. Moreover, established theory deals in qualitative measures when describing the optimal datasets for Hand-Eye calibration when the system can be commanded to take any pose [6, 7], but leaves something to be desired when it comes to deciding how datapoints should be picked and paired in a dataset consisting of a possibly very large amount of data.

With a motivation to answer these questions and simultaneously develop methods to more easily analyse the Hand-Eye calibration problem, this thesis develops new theoretical frameworks. The derived theory enables the quantification of *information* present in any single datapoint to be used in Hand-Eye calibration, relative the entirety of a dataset. This is done by analyzing a specific Hand-Eye solver with properties advantageous for such developments, the Park-Martin solver. Enabled by the theoretical derivations in this thesis, methods are shown capable of estimating the camera orientation to within 1.3° of the ground-truth values, and performing the estimation accurately with 3 times fewer datapoints than previously demonstrated. The theoretical framework enabled also opens the door to interesting questions to pursue in future work.

The problem of estimating the position of a ship-mounted camera is not considered neither in this thesis nor the spesialization project, due to this problem still being difficult to solve for ship-like movements.

1.1 Organization

This thesis is organized into parts roughly defined as follows:

- Introductory material and existing theory in literature, Chapters 1 and 2.
- New theoretical derivations done in this thesis, Sections 3.1 to 3.3.
- Proposed methodology, Sections 3.3.5 and 3.4.
- Simulation results testing derived theory and proposed methodology, Chapter 4.
- Discussion and conclusion of results in this thesis, Chapters 5 and 6.

Chapter 2

Theory

2.1 Coordinate frames

Coordinate frames are used to represent the poses of rigid objects as well as describing the transformation of vectors between multiple rigid objects. For 3-dimensional space, defining the coordinate system A is done by defining three orthonormal vectors, or axes, $(\mathbf{x}_\mathrm{A}, \mathbf{y}_\mathrm{A}, \mathbf{z}_\mathrm{A})$ centered at an origin $\mathcal{O}_\mathrm{A}$. With this, any point may be defined relative to frame A as a unique linear combination of the three axes.

Further, if three new orthonormal vectors are defined as a linear combination of the coordinate axes of frame A one may define a second coordinate frame. Naming this coordinate system B, defining its origin $\mathcal{O}_\mathrm{B}^\mathrm{A}$ and collecting its axes into the columns of a matrix $\mathbf{R}_\mathrm{AB}$ as in Equation (2.1), we are able to define the orientation and position of B relative to A numerically.

$$\mathbf{R}_\mathrm{AB} = \begin{bmatrix} | & | & | \\ \mathbf{x}_\mathrm{B}^\mathrm{A} & \mathbf{y}_\mathrm{B}^\mathrm{A} & \mathbf{z}_\mathrm{B}^\mathrm{A} \\ | & | & | \end{bmatrix} \tag{2.1}$$

2.1.1 Notation

In this thesis, the following notation is adhered to when it comes to the notation of coordinate frames and similar mathematical objects.

- Any non-scalar object is given in bold. $\mathbf{v}, \mathbf{A}$.
- Vectors are written in lowercase, matrices in uppercase.
- When relevant, the coordinate system for which a vector is defined in is superscripted. $\mathbf{v}^\mathrm{a}$. When not relevant, this is omitted.
- Coordinate transforms are given on the form $\mathbf{H}_\mathrm{ab}$, being understood as either a matrix finding the coordinate expression in coordinate system “a” of a vector given in coordinate system “b”, or as the pose of coordinate system “b” relative system “a”.

- The angle-axis representation of orientations is denoted with θ as the angle, $\mathbf{a}$ as the unit-norm axis, and $\boldsymbol{\omega} = \theta\mathbf{a}$ as the resultant rotation vector.

Coordinate frames and their origins then allow for mathematical description of the relative orientation and position of rigid objects. This is done by defining coordinate frames that are fixed to the geometry of these objects, following some chosen convention. The following describes different such conventions for defining the coordinate frames that are relevant for this project.

2.1.2 The Body Frame

For marine vessels a common practice when defining a coordinate system rigidly attached to the ship is to define the X-axis pointing forwards along the bow, the Z-axis to be pointed downwards and the Y-axis to complete the right-handed coordinate system [8]. This coordinate frame is simply dubbed the *body frame*.

2.1.3 The Camera Frame

Some users [9–11] prefer to define the Z-axis of cameras to point along the optical axis, the Y-axis to point downwards along the camera body and the X-axis to complete the right-handed system. Others, however, prefer to have the X-axis be pointed along the optical axis, the Z-axis pointing upwards, and the Y-axis thereafter.

Naming the conventions “A” (Z along optical axis, Y down) and “B” (X along optical axis, Z up) respectively, Equation (2.2) relates the two through a rotation matrix.

$$\mathbf{R}_{\mathrm{AB}} = \begin{bmatrix} \mathbf{x}_{\mathrm{B}}^{\mathrm{A}} & \mathbf{y}_{\mathrm{B}}^{\mathrm{A}} & \mathbf{z}_{\mathrm{B}}^{\mathrm{A}} \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & -1 \\ 1 & 0 & 0 \end{bmatrix} \tag{2.2}$$

For marine operations where the body-coordinate system often is defined with the X-axis forwards and the Z-axis downwards, some might find it intuitive to define the camera coordinate frame equivalently. Therefore, a third convention is to have the X-axis pointing along the optical axis, the Z-axis pointing downwards and the Y-axis completing the coordinate system. The transformation relating this convention, “C”, and convention “A” is given in Equation (2.3). An illustration of all three common camera frame conventions is given in Figure 2.1.

$$\mathbf{R}_{\mathrm{AC}} = \begin{bmatrix} \mathbf{x}_{\mathrm{C}}^{\mathrm{A}} & \mathbf{y}_{\mathrm{C}}^{\mathrm{A}} & \mathbf{z}_{\mathrm{C}}^{\mathrm{A}} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \tag{2.3}$$

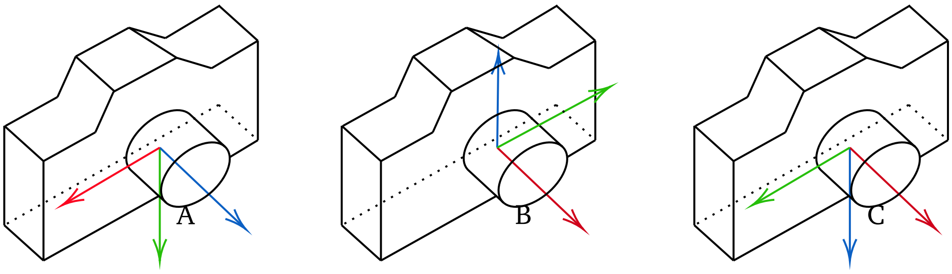


Figure 2.1: Three common conventions for defining a coordinate frame rigidly attached to a camera. In the figure, standard coloring of the axes as red=X, green=Y and blue=Z is used.

2.1.4 North-East-Down inertial frame

While the body- and camera-frame are frames attached to objects to describe their pose, the North-East-Down (NED) coordinate frame is a commonly used inertial frame for these objects to be positioned relative to.

The globe is near spherical and therefore navigational data will reflect this. This can be impractical when using motion models and inertial navigation, which often assume the world to be locally flat. The NED coordinate system is constructed by choosing a reference point at which a plane tangent to the globe is constructed. Then local to this point, other measured ship-positions will appear planar and so the assumption holds [8].

2.2 The SO(3) group

When three orthonormal axes are ordered as columns in a matrix the resulting matrix is called orthonormal as well. The set of all orthonormal matrices that have its determinant equal to 1 form a group under matrix multiplication $(\cdot)$, called the *special orthogonal* group of dimension 3: SO(3). The group may be defined as in Equation (2.4).

$$\mathrm{SO}(3) = \left(\{\mathbf{R} \in \mathbb{R}^{3\times 3} \mid \mathbf{R}^\top \mathbf{R} = \mathbf{R}\mathbf{R}^\top = \mathbf{I}_{3\times 3},\ \det(\mathbf{R}) = +1\},\ (\cdot)\right) \tag{2.4}$$

Elements of SO(3) are to be understood as the *orientation* of objects, or the *rotation* of vectors between objects. The rotation matrices have 9 elements, but only 3 degrees of freedom [12]. Therefore multiple simpler parametrizations of SO(3) exist rather than defining three orthonormal vectors. Most notable are the angle-axis representation, Euler-angles and quaternions [8].

2.2.1 Lie groups

Since the set of SO(3) is defined by all 3 by 3 matrices satisfying a constraint, then this set is by definition a manifold on $\mathbb{R}^{3\times 3}$ defined by said constraint. The constraint can be shown to be differentiable, which classifies SO(3) as a *Lie group* [12].

Lie groups are special in that their differentiable nature allows for the existence of tangent spaces centered at any group element, where these tangent spaces are n-dimensional vector spaces. More importantly, it is possible to define an *exponential map* that maps vectors in the tangent space to group elements. The tangent space of a group $\mathcal{G}$ at its identity element $\mathcal{E}$ is its associated *Lie algebra*, denoted $\mathfrak{g}$, which in itself has properties not used in this thesis directly. The fact that these tangent spaces and the Lie algebra in particular are real vector spaces makes them isomorphic to $\mathbb{R}^n$. This allows for easy application of useful and familiar concepts over the vector space $\mathbb{R}^n$, such as convexity and the derivative, onto the much more complicated group-structure simply by casting elements to and from the Lie algebra using the aforementioned exponential map. The functions transforming elements between the Lie group, the Lie algebra and vectors in $\mathbb{R}^n$ are here denoted by the symbols given in Equations (2.5) to (2.8), inspired by the work of Solà *et al.* [12]. Equation (2.5) is the aformentioned exponential map and Equation (2.6) is its inverse, while Equation (2.7) is the *vee* function and Equation (2.8) is its inverse the *wedge* function.

$$\exp : \mathfrak{g} \rightarrow \mathcal{G} \tag{2.5}$$

$$\log : \mathcal{G} \rightarrow \mathfrak{g} \tag{2.6}$$

$$(\cdot)^{\vee} : \mathfrak{g} \rightarrow \mathbb{R}^n \tag{2.7}$$

$$(\cdot)^{\wedge} : \mathbb{R}^n \rightarrow \mathfrak{g} \tag{2.8}$$

Solà *et al.* additionally define short-hand functions for transforming directly between the Lie group and the reals, shown in Equations (2.9) and (2.10).

$$\mathrm{Exp} : \mathbb{R}^n \rightarrow \mathcal{G}, \quad \mathrm{Exp}(\mathbf{a}) = \exp(\mathbf{a}^{\wedge}) \tag{2.9}$$

$$\mathrm{Log} : \mathcal{G} \rightarrow \mathbb{R}^n, \quad \mathrm{Log}(\mathbf{R}) = \log(\mathbf{R})^{\vee} \tag{2.10}$$

If the group $\mathcal{G}$ in Equations (2.5) to (2.8) is the Lie group SO(3), its Lie algebra is denoted $\mathfrak{so}(3)$. The elements of $\mathfrak{so}(3)$ are skew-symmetric matrices and associating these with vectors in $\mathbb{R}^3$ is shown in Equation (2.11).

$$\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix}^{\wedge} = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} \tag{2.11}$$

Interestingly, the corresponding column vector of some element in $\mathfrak{so}(3)$ is exactly the well-known rotation vector of the associated rotation. This last property comes from the fact that the exponential map $\exp : \mathfrak{so}(3) \rightarrow \mathrm{SO}(3)$ is exactly the Rodrigues rotation formula, seen in Equation (2.12). Here, $[\cdot]_\times$ is principally the same action as $(\cdot)^{\wedge}$, sending vectors to skew-symmetric matrices.

$$\exp(\boldsymbol{\omega}^{\wedge}) = \mathbf{I}_{3\times3} + \sin(\theta)\,[\mathbf{a}]_\times + (1 - \cos(\theta))\,[\mathbf{a}]_\times^2 \tag{2.12}$$

For the SO(3) group, the inverse of the exponential map is the logarithm map seen in Equation (2.13).

$$\log(\mathbf{R}) = \frac{\theta(\mathbf{R}-\mathbf{R}^\top)}{2\sin(\theta)}, \text{ where } \theta = \arccos\left(\frac{\operatorname{tr}(\mathbf{R})-1}{2}\right) \in [0,\pi) \tag{2.13}$$

Two more useful properties of the SO(3) group are listed in Proposition 1 and Proposition 2. These properties are taken from the work of Park and Martin [13], for which the authors provide proofs.

Proposition 1. *Let* $\mathbf{B},\mathbf{X} \in \mathrm{SO}(3)$*, then* $\log\left(\mathbf{X}\mathbf{B}\mathbf{X}^\top\right) = \mathbf{X}\log(\mathbf{B})\mathbf{X}^\top$

Proposition 2. *Let* $\mathbf{B},\mathbf{X} \in \mathrm{SO}(3)$*, then* $\mathbf{X}\log(\mathbf{B})\mathbf{X}^\top = (\mathbf{X}\mathrm{Log}(\mathbf{B}))^\wedge$

It must be noted that the theory of Lie groups is much more complex than presented here, and is simplified for the purposes of this thesis.

2.2.2 Representing noise over a rotation

This chapter is largely a reproduction of the theory presented by Mangelson *et al.* in [14].

Observations and measurements are intrinsically uncertain. Representing this uncertainty over a vector of measurements is often done by assuming the measurement is *perturbed* by some small stochastic variable. The uncertainty can then be modelled by choosing an appropriate probability density function for the stochastic variable, often the multivariate Gaussian distribution. The multivariate Gaussian is parametrized by a positive semi-definite *covariance matrix* $\boldsymbol{\Sigma} \in \mathbb{R}^{N\times N}$ and *mean* $\boldsymbol{\mu} \in \mathbb{R}^N$, the latter of which is often set to zero when modelling measurement noise as perturbations. This is described in Equation (2.14), where $\bar{\mathbf{y}}$ is the mean or "true" value being measured and $\mathbf{z}$ is the small perturbation by noise.

$$\mathbf{y} = \begin{bmatrix} \bar{\mathbf{y}}_1 \\ \vdots \\ \bar{\mathbf{y}}_N \end{bmatrix} + \begin{bmatrix} \mathbf{z}_1 \\ \vdots \\ \mathbf{z}_N \end{bmatrix}, \text{ where } \mathbf{z} := \begin{bmatrix} \mathbf{z}_1 \\ \vdots \\ \mathbf{z}_N \end{bmatrix} \sim \mathcal{N}(\mathbf{0},\boldsymbol{\Sigma}) \tag{2.14}$$

For tasks where the pose of some object is of interest, it is natural to want to describe the uncertainty of said pose. It has been shown that doing so in a way that preserves the group-structure of the poses leads to better results than simply modelling noise over for example the Euler-angles [15].

The method is as follows. We once more consider measurements to be perturbed by some Gaussian noise-vector, $\xi \sim \mathcal{N}(\mathbf{0},\boldsymbol{\Sigma})$. The noise is then applied to the *mean pose* $\overline{\mathbf{H}}$ by multiplying with its exponentiation, either to the left or the right; $\mathbf{H} = \exp(\xi^\wedge)\cdot\overline{\mathbf{H}}$ or $\mathbf{H} = \overline{\mathbf{H}}\cdot\exp(\xi^\wedge)$. The perturbation is seen as an element of the group tangent space, where the difference between left- and right multiplication signifies whether the perturbation is part of the Lie algebra centered at the identity element or the tangent space centered at $\overline{\mathbf{H}}$, respectively. These can be understood as whether the perturbations have coordinates given relative the *global* or *local* reference frames. In this thesis, the latter will be used as this leads to easier interpretation of the covariances, but the former is equally as valid.

Other authors [14, 16] expand on this model to show how one can perform propagation of correlation using this method. This allows for an even more accurate representation of the uncertainty over poses, especially when one is interested in the composition of multiple poses.

2.2.3 A metric over the group of 3D rotations

A *metric* is a real-valued function defined over some set which enables a notion of closeness between elements of the set. Sets for which such a function is defined is called a *metric space*, a structure useful in its own right. In this thesis, defining a metric over SO(3) will be useful for comparing multiple orientation-estimates against a ground truth.

Many metrics may be defined over SO(3), but one metric of particular geometric interpretation is the one used in [13], restated in Equation (2.15). Here, $||\cdot||_2$ is the standard Euclidian norm of a vector.

$$d(\mathbf{A},\mathbf{B}) = ||\log(\mathbf{A}^\top \mathbf{B})^\vee||_2,\ \mathbf{A},\mathbf{B} \in \mathrm{SO}(3) \tag{2.15}$$

The metric may be understood as the angle of the shortest rotation connecting the orientations **A** and **B**, or equivalently the shortest path over $\mathfrak{so}(3)$ connecting the two elements [17].

2.2.4 Other properties

Lastly, two useful operations over SO(3) used later are presented briefly.

Firstly, with the aforementioned notion of "perturbing a group element" it is possible to define a notion of the derivative of functions over a group. This is done as

$$\frac{\partial f(\mathcal{X})}{\partial \mathcal{X}} := \lim_{\boldsymbol{\tau}\to \mathbf{0}} \frac{\mathrm{Log}(f(\mathcal{X})^{-1} \circ f(\mathcal{X} \circ \mathrm{Exp}(\boldsymbol{\tau})))}{\boldsymbol{\tau}}. \tag{2.16}$$

In this way, one can define Jacobians for commonly used functions of rotations [12]. These Jacobians also allow for the definition of Taylor expansions and thereby the propagation of noise through nonlinear functions.

Two particularly useful Jacobians of functions over SO(3) used in this thesis are the *group action* derivative, Equation (2.17), and *right Jacobian*, Equation (2.18).

$$\mathbf{J}_{\mathbf{R}}^{\mathbf{R}\boldsymbol{\beta}} = -\mathbf{R}[\boldsymbol{\beta}]_\times \tag{2.17}$$

$$\mathbf{J}_r(\theta\mathbf{a}) := \mathbf{J}_{\boldsymbol{\omega}}^{\exp(\boldsymbol{\omega}^\wedge)} = \mathbf{I}_{3\times 3} - \frac{1-\cos(\theta)}{\theta^2}[\mathbf{a}]_\times + \frac{\theta - \sin(\theta)}{\theta^3}[\mathbf{a}]_\times^2 \tag{2.18}$$

Secondly, the Baker–Campbell–Hausdorf (BCH) formula seen in Equation (2.19) allows for useful association of two Lie algebra elements with their exponential [16]. Here, $[\cdot,\cdot]$ is the *Lie bracket* of the relevant Lie algebra, which for $\mathfrak{so}(3)$ is the commutator for matrices $[\mathbf{A},\mathbf{B}] = \mathbf{AB} - \mathbf{BA}$.

$$\begin{aligned}\log(\exp(\boldsymbol{\omega}_1^\wedge)\exp(\boldsymbol{\omega}_2^\wedge)) = \boldsymbol{\omega}_1^\wedge + \boldsymbol{\omega}_2^\wedge + \frac{1}{2}[\boldsymbol{\omega}_1^\wedge, \boldsymbol{\omega}_2^\wedge] \\ + \frac{1}{12}[\boldsymbol{\omega}_1^\wedge, [\boldsymbol{\omega}_1^\wedge, \boldsymbol{\omega}_2^\wedge]] \\ + \frac{1}{12}[\boldsymbol{\omega}_2^\wedge, [\boldsymbol{\omega}_2^\wedge, \boldsymbol{\omega}_1^\wedge]]\dots\end{aligned} \tag{2.19}$$

More importantly, taking the *vee*-function of each side of Equation (2.19) reveals the important approximate relationship

$$\mathrm{Exp}(\boldsymbol{\omega}_1)\mathrm{Exp}(\boldsymbol{\omega}_2) \approx \mathrm{Exp}(\boldsymbol{\omega}_1 + \boldsymbol{\omega}_2), \tag{2.20}$$

an approximation whose error is small if either $\boldsymbol{\omega}_1$ or $\boldsymbol{\omega}_2$ is a small vector. The approximation becomes an equality if the $\mathfrak{so}(3)$ elements commute, which for their $\mathbb{R}^3$ equivalents mean they are parallel.

2.3 The SE(3) group and Homogeneous Transforms

As introduced in Section 2.1, describing the pose of objects relative each other is done with an orientation and a position. In Section 2.2, the mathematical properties of such orientations is explained through the language of rotation matrices $\mathbf{R}$. The position of an object in space is trivially described by a three-dimensional vector $\mathbf{t}$. The pair $(\mathbf{R}, \mathbf{t})$ is then a mathematical description of a pose.

Similarly to rotations, poses can act as an action over vectors in addition to being seen as objects in themselves. Letting $\mathbf{T}_{\mathrm{nb}} = (\mathbf{R}_{\mathrm{nb}}, \mathbf{t}_{\mathrm{nb}})$ be the pose of some coordinate frame "b" relative the coordinate frame "n", then the action of $\mathbf{T}_{\mathrm{nb}}$ over a vector expressed in the former frame gives the same vector expressed in the latter frame. This is done mathematically as seen in Equation (2.21).

$$\mathbf{T}_{\mathrm{nb}} \cdot \mathbf{p}^{\mathrm{b}} = \mathbf{R}_{\mathrm{nb}}\mathbf{p}^{\mathrm{b}} + \mathbf{t}_{\mathrm{nb}} = \mathbf{p}^{\mathrm{n}} \tag{2.21}$$

The action of poses over vectors can be represented more elegantly through the use of Homogeneous Transformation (HT) matrices. These are 4 × 4 real matrices consisting of both the rotation matrix $\mathbf{R}$ and position vector $\mathbf{t}$, seen in Equation (2.22). Constructing a homogeneous coordinate vector, $\tilde{\mathbf{p}}^b = \begin{bmatrix}\mathbf{p}^{b\top} & 1\end{bmatrix}^\top$, the group action of SE(3) over the homogeneous coordinate vectors is simply matrix multiplication $\mathbf{H}_{\mathrm{nb}}\tilde{\mathbf{p}}^b = \tilde{\mathbf{p}}^n$ [11].

$$\mathbf{H}_{\text{nb}} = \begin{bmatrix} \mathbf{R}_{\text{nb}} & \mathbf{t}_{\text{nb}} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix} \tag{2.22}$$

$$\mathbf{H}_{\text{nb}}^{-1} = \mathbf{H}_{\text{bn}} = \begin{bmatrix} \mathbf{R}_{\text{nb}}^{\top} & -\mathbf{R}_{\text{nb}}^{\top}\mathbf{t}_{\text{nb}} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix} \tag{2.23}$$

Homogeneous Transformation matrices are commonly used as a representation of the pose both in the fields of computer vision [10, 11] and robotics [18].

Poses are elements of the rigid motion group SE(3), and therefore HTs are a representation of SE(3)-elements. The group of rigid motions is a Lie group similarly to SO(3). This means one can identify the Lie algebra by the group's tangent space and thereby perform vector operations that are mapped to the group through the exponential map. In this thesis, only the orientation of objects is estimated, and as such properties of the SE(3) Lie algebra is not utilized. A further explanation is therefore omitted.

2.3.1 Relative pose

The following explanation given below is in large taken from the specialization project written by the author.

The concept of a *relative pose* is used extensively throughout this project to describe data and its properties. Relative pose should be understood as the following. Poses defined in some inertial frame are said to be *absolute poses*. Let $\mathbf{H}_{\text{na}}$ and $\mathbf{H}_{\text{nb}}$ be HTs describing the pose of two different frames, "a" and "b", relative the same inertial frame, "n". The relative pose of b relative a is then computed as Equation (2.24). Figure 2.2 illustrates the interpretation of relative pose, as defined in this project.

$$\begin{aligned} \mathbf{H}_{\text{ab}} &= \mathbf{H}_{\text{na}}^{-1}\mathbf{H}_{\text{nb}} \\ &= \mathbf{H}_{\text{an}}\mathbf{H}_{\text{nb}} \end{aligned} \tag{2.24}$$

2.4 The approximate Hessian in nonlinear least squares

The Hessian matrix, or simply *Hessian*, is the matrix of second-derivatives of some scalar function. The Hessian is of importance in optimization and estimation, as it conveys information about the function's convexity and is related to the covariance of estimates [19]. More fundamentally, the Hessian shows up in the Taylor-expansion of a scalar field. Equation (2.25) shows the Taylor-expansion centered at $\mathbf{x}_0$ of a scalar field $f(\mathbf{x})$. Here $\mathbf{J}_0 := \mathbf{J}_{\mathbf{x}}^{f}(\mathbf{x}_0) = \nabla_{\mathbf{x}} f^{\top}(\mathbf{x}_0)$ is the Jacobian of $\mathbf{f}$ with respect to $\mathbf{x}$ evaluated at the center of the Taylor expansion, $\mathbf{x}_0$, while $\mathbf{H}_0 := \mathbf{J}_{\mathbf{x}}^{\nabla_{\mathbf{x}} f}(\mathbf{x}_0)$ is the Hessian evaluated at the center.

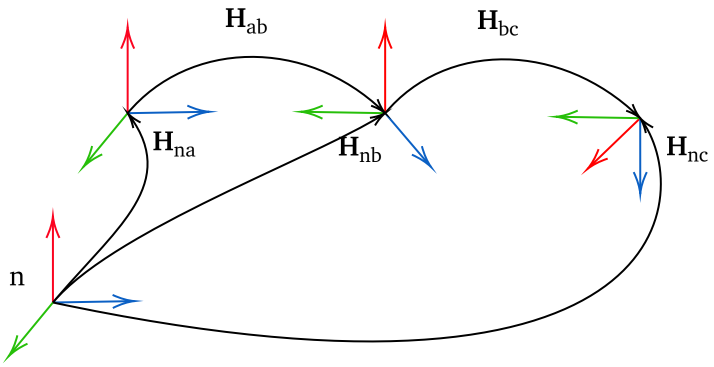


Figure 2.2: The relative pose between three coordinate frames, and their relationship with the reference frame

$$f(\mathbf{x}) \approx f(\mathbf{x}_0) + \mathbf{J}_0 \cdot (\mathbf{x} - \mathbf{x}_0) + \frac{1}{2}(\mathbf{x} - \mathbf{x}_0)^\top \mathbf{H}_0(\mathbf{x} - \mathbf{x}_0) + ... \tag{2.25}$$

The scalar field $f(\mathbf{x})$ is said to be convex everywhere if its Hessian is positive semi-definite. This applies to the Taylor-expansion as well, and if an n-th order approximation of f is constructed by ignoring every term beyond the $n+1$-th of Equation (2.25), then the approximation is *locally convex* around $\mathbf{x}_0$ if $\mathbf{H}_0$ is positive semi-definite[20].

In nonlinear least squares, the cost-function to be minimized is such a scalar field. This is often done by minimizing the sum-of-squared residuals, seen in Equation (2.26). Here, $\mathbf{y} = [\mathbf{y}_1^\top, \ldots, \mathbf{y}_N^\top]^\top$ is the vector of measured values, $\mathbf{f}(\mathbf{x}) = [\mathbf{f}_1(\mathbf{x})^\top, \ldots, \mathbf{f}_N(\mathbf{x})^\top]^\top$ is the vector of predicted values for some input $\mathbf{x}$ and $\mathbf{r} = \mathbf{y} - \mathbf{f}(\mathbf{x})$ is the vector of residuals.

$$\min_{\mathbf{x}} F(\mathbf{x}) = \min_{\mathbf{x}} \frac{1}{2} \sum_{i=1}^{N} ||\mathbf{y}_i - \mathbf{f}_\mathbf{i}(\mathbf{x})||^2 = \min_{\mathbf{x}} \frac{1}{2} ||\mathbf{y} - \mathbf{f}(\mathbf{x})||^2 = \min_{\mathbf{x}} \frac{1}{2} \mathbf{r}^\top \mathbf{r} \tag{2.26}$$

When performing nonlinear least squares, many methods make use an estimate of the Hessian of the cost-function to be minimized in Equation (2.26). This will for large optimization problems be infeaseable, due to the amount of variables this would incur. A common approach is therefore to perform an approximation of the Hessian of the cost-function. The derivation of which is reproduced below.

Taking the first-order Taylor expansion of the measurement-prediction function $\mathbf{f}$ around some chosen point $\mathbf{x}_0$ would lead to the form $\mathbf{f}(\mathbf{x}) \approx \mathbf{f}(\mathbf{x}_0) + \mathbf{J}_0 \cdot (\mathbf{x} - \mathbf{x}_0)$. Note how the Jacobian, $\mathbf{J}_\mathbf{x}^\mathbf{f}(\mathbf{x}_0)$, now is a matrix since $\mathbf{f}$ is vectorial, as opposed to being a vector as in Equation (2.25). Choosing $\mathbf{x}_0$ such that $\mathbf{f}(\mathbf{x}_0) = \mathbf{y}$, which means centering the Taylor expansion at the global minimum of the optimization problem, the optimization problem simplifies to Equation (2.27).

$$\min_{\mathbf{x}} \frac{1}{2} ||\mathbf{J}_0 \cdot (\mathbf{x} - \mathbf{x}_0)||^2 \tag{2.27}$$

Naturally, the value of $\mathbf{x}_0$ such that $\mathbf{f}(\mathbf{x}_0) = \mathbf{y}$ is not known, that is the whole point of iterative optimization, but the form in Equation (2.27) does lead to easier analysis onwards. Notably, defining $\tilde{F}(\mathbf{x}) := \frac{1}{2}||\mathbf{J}_0 \cdot (\mathbf{x} - \mathbf{x}_0)||^2$ as an approximation of the full optimization function, the Hessian of the actual cost-function $F(\mathbf{x})$ in Equation (2.26) can be approximated by Equation (2.28).

$$\begin{aligned}
\nabla_{\mathbf{x}} \tilde{F}(\mathbf{x}) &= \nabla_{\mathbf{x}} \frac{1}{2} ||\mathbf{J}_0 \cdot (\mathbf{x} - \mathbf{x}_0)||^2 && (2.28)\\
&= \nabla_{\mathbf{x}} \frac{1}{2} (\mathbf{J}_0 \cdot (\mathbf{x} - \mathbf{x}_0))^\top (\mathbf{J}_0 \cdot (\mathbf{x} - \mathbf{x}_0)) && (2.29)\\
&= \nabla_{\mathbf{x}} \frac{1}{2} (\mathbf{x} - \mathbf{x}_0)^\top (\mathbf{J}_0^\top \mathbf{J}_0)(\mathbf{x} - \mathbf{x}_0) && (2.30)\\
&= (\mathbf{J}_0^\top \mathbf{J}_0)(\mathbf{x} - \mathbf{x}_0) && (2.31)
\end{aligned}$$

$$\begin{aligned}
\nabla^2_{\mathbf{x}} \tilde{F}(\mathbf{x}) &= \nabla_{\mathbf{x}} (\mathbf{J}_0^\top \mathbf{J}_0)(\mathbf{x} - \mathbf{x}_0) && (2.32)\\
&= \mathbf{J}_0^\top \mathbf{J}_0 && (2.33)
\end{aligned}$$

These results show how the Hessian of the cost function can be approximated by the square of the Jacobian of the prediction-functions. Further, since the first order Taylor approximation of **f** is exactly equal to the full function when evaluated *at* the centre of the expansion, the apporoximated Hessian will also be "perfect" when evaluated at the global minima. When using this approximation in optimization, the Jacobian evaluated at the minima is not known since the minima is unknown. This is solved by instead developing the approximate Hessian about the current estimate of the minima, and finding the step of steepest descent away from this estimate [20].

2.5 Egomotion estimation algorithms

Multiple algorithms have been developed in the field of computer vision that produce an estimate of the movement of a camera given a set of pictures. Such methods are unified under the term "camera egomotion estimation" [21], and some famous examples include methods of Simultaneous Localization and Mapping (SLAM), Visual Odometry (VO), and Structure from Motion (SfM). Explained briefly, estimating camera egomotion is often done by tracking the movement of points in the real world as observed through the subsequent images for which those points are visible [22]. Given the tracks of these points across multiple images, the camera motion relative these points can be estimated given geometric or numeric considerations [4, 23]. Lastly, these rough motion estimates are refined by softly enforcing some constraint, often by some optimization. If the scale of the environment is known, for example by identifying the known distance between two estimated points or using calibration plates, then the estimated camera motions can also be given in this same scale. Using calibration plates does however require additional infrastructure, which is not always possible. If, however, such scale information is not present then the camera motions will be given in an arbitrary scale not necessarily equal to the metric scale of the performed motions.

2.6 Hand-Eye calibration

The *Hand-Eye calibration problem* originates in robotics and concerns the issue of finding how a sensor capable of egomotion estimation, often a camera, is rigidly mounted relative some end-effector whose pose can be controlled. The parameters to be estimated are the *extrinsic parameters* describing said relationship. Study of the Hand-Eye calibration problem formulation is often attributed to being done first by Shiu *et al.* in 1989 [1]. Since then many papers have been written on different solution techniques to recover the extrinsic parameters [5, 13, 24], with research mainly focused on improving the accuracy given better computational power the last few decades.

2.6.1 Visual derivation

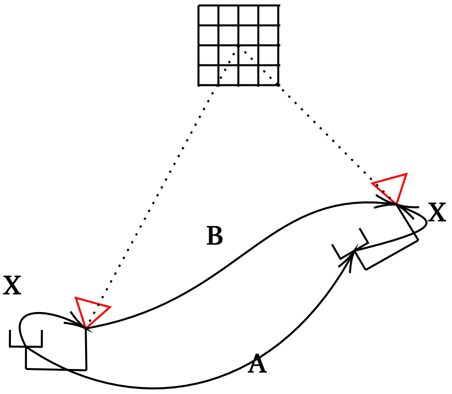


Figure 2.3: Conceptual illustration of the Hand-Eye Calibration problem. The setup consists of a camera (red), an end-effector and a calibration plate. In the illustration, the rigid system undergoes some controlled motion.

To derive the problem formulation, consider Figure 2.3. In the original formulation of the calibration problem, a rig consisting of a camera and end-effector is attached to a robotic arm, which in turn allows for precise movement of the end-effector. The setup is moved between two predetermined poses and pictures of a stationary calibration plate are captured at each pose. The relative pose of the camera between each picture may then be calculated by any of a multitude of egomotion algorithms, and since the dimensions of such a calibration pattern are known then the translation part of the relative pose is known to correct metric scale. The relative pose of the end-effector is naturally also known since the whole point of a robotic arm is the precise control of the end-effector. Assuming the mounting of the camera relative the hand is constant for the timespan analyzed then by visual proof the following equivalence can be made (a rigorous proof will follow in Section 2.6.2):

$$\mathbf{AX} = \mathbf{XB} \tag{2.34}$$

Here, **A** is the Homogeneous Transformation describing the relative pose of the end-effector between the two pictures, **B** is similarly the relative pose of the camera, and **X** is the assumed constant HT of the extrinsics. Following the arrows in Figure 2.3, Equation (2.34) follows naturally.

2.6.2 Formulation for ship-data

In previous work by the author, it has been shown that it is possible to fit the Hand-Eye calibration problem formulation of robotic hands onto the case of ship-mounted cameras and thereby enable finding the extrinsics of the ship-mounted cameras through any technique which works on the robot-arm setup. The following derivation is in part taken from a conference proceeding written by the author, to be published at the Fusion 2023 conference in June of 2023. A preprint of the paper can be seen in Appendix C.

Modern ships are equipped with advanced sensor-suites which fuse GNSS measurements with inertial- and attitude measurements. This means the ship's absolute pose in the NED coordinate frame is available frequently and with high accuracy. If such a ship is equipped with a camera then any egomotion algorithm presented in Section 2.5 can be used to reconstruct the movement of the camera relative an unknown reference frame chosen arbitrarily by the algorithm, although with unknown scale.

Let $\mathbf{H}_{\mathrm{nb}}(t_j)$ be the HT describing the measured pose of the ship body frame "b" relative the chosen NED frame "n" at some timestamp t_j. Let $\mathbf{H}_{\mathrm{ni}}(t_j)$ be the reconstructed egomotion of the camera frame "i" relative NED at the same timestamp. Note that this last measurement is not available, since the camera reconstruction is given relative some unknown mediary frame denoted "m" used by the egomotion algorithm. This will be addressed below. Using these symbols, the unknown extrinsic calibration is the HT $\mathbf{H}_{\mathrm{bi}}(t_p) = \mathbf{H}_{\mathrm{bi}}(t_q) := \mathbf{H}_{\mathrm{bi}},\ \forall t_p,\ t_q$. With these mathematical symbols defined, the following derivation can be performed. Readers are advised to keep close attention to subscripts and to the difference between left- and right-multiplication of matrices.

$$\begin{aligned}
\mathbf{I}_{4\times4} &= \mathbf{I}_{4\times4} \\
\mathbf{G}\mathbf{G}^{-1} &= \mathbf{H}\mathbf{H}^{-1}, && ,\forall\ \mathbf{G},\mathbf{H}\in\mathrm{SE}(3) \\
\mathbf{H}_{\mathrm{nb}}(t_q)\mathbf{H}_{\mathrm{bn}}(t_q) &= \mathbf{H}_{\mathrm{nb}}(t_p)\mathbf{H}_{\mathrm{bn}}(t_p) && ,\forall\ t_p\neq t_q \\
\mathbf{H}_{\mathrm{nb}}(t_q)\mathbf{H}_{\mathrm{bi}}(t_q)\mathbf{H}_{\mathrm{in}}(t_q) &= \mathbf{H}_{\mathrm{nb}}(t_p)\mathbf{H}_{\mathrm{bi}}(t_p)\mathbf{H}_{\mathrm{in}}(t_p) && ,\mathbf{H}_{\mathrm{bn}}(t_p)\cdot \\
\mathbf{H}_{\mathrm{bn}}(t_p)\mathbf{H}_{\mathrm{nb}}(t_q)\mathbf{H}_{\mathrm{bi}}\mathbf{H}_{\mathrm{in}}(t_q) &= \mathbf{H}_{\mathrm{bi}}\mathbf{H}_{\mathrm{in}}(t_p) && ,\cdot\ \mathbf{H}_{\mathrm{ni}}(t_q) \\
\mathbf{H}_{\mathrm{bn}}(t_p)\mathbf{H}_{\mathrm{nb}}(t_q)\mathbf{H}_{\mathrm{bi}} &= \mathbf{H}_{\mathrm{bi}}\mathbf{H}_{\mathrm{in}}(t_p)\mathbf{H}_{\mathrm{ni}}(t_q) \\
\mathbf{H}_{\mathrm{nb}}^{-1}(t_p)\mathbf{H}_{\mathrm{nb}}(t_q)\mathbf{H}_{\mathrm{bi}} &= \mathbf{H}_{\mathrm{bi}}\mathbf{H}_{\mathrm{ni}}^{-1}(t_p)\mathbf{H}_{\mathrm{ni}}(t_q)
\end{aligned} \tag{2.35}$$

The last line of this derivation is very similar to the Hand-Eye calibration problem, but two challenges must be addressed. These being that neither $\mathbf{H}_{\mathrm{ni}}^{-1}(t_p)$ nor $\mathbf{H}_{\mathrm{ni}}(t_q)$ are known and the fact that the actual measurement has translation given in some unknown scale, due to the landmark-based camera egomotion reconstruction. The actual measurements $\mathbf{H}_{\mathrm{mi}}(t_j)$ are related to these unknown HTs through the relationship $\mathbf{H}_{\mathrm{ni}}(t_j) = \mathbf{H}_{\mathrm{nm}}\mathbf{H}_{\mathrm{mi}}^{\lambda}(t_j)$, where $\mathbf{H}_{\mathrm{mi}}^{\lambda}(t_j) := \mathbf{f}_{\lambda}(\mathbf{H}_{\mathrm{mi}}(t_j))$ is the function sending Homogeneous Transformation matrices to their scaled variant, and λ is the unknown scale factor.

$$\mathbf{f}_{\lambda} : \mathrm{SE}(3) \to \mathrm{SE}(3),\ \mathbf{f}_{\lambda}(\mathbf{H}) = \begin{bmatrix} \mathbf{R} & \lambda\mathbf{t} \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix},\ \lambda\in\mathbb{R}_{>0} \tag{2.36}$$

Then one can see that

$$\mathbf{H}_{\mathrm{ni}}^{-1}(t_p)\mathbf{H}_{\mathrm{ni}}(t_q) = \left(\mathbf{H}_{\mathrm{nm}}\mathbf{H}_{\mathrm{mi}}^{\lambda}(t_p)\right)^{-1}\mathbf{H}_{\mathrm{nm}}\mathbf{H}_{\mathrm{mi}}^{\lambda}(t_q) = \left(\mathbf{H}_{\mathrm{mi}}^{\lambda}(t_p)\right)^{-1}\mathbf{H}_{\mathrm{mi}}^{\lambda}(t_q) \tag{2.37}$$

where this last equality becomes

$$\begin{aligned}
\left(\mathbf{H}_{\mathrm{mi}}^{\lambda}(t_p)\right)^{-1}\mathbf{H}_{\mathrm{mi}}^{\lambda}(t_q) &= \left(\mathbf{f}_{\lambda}(\mathbf{H}_{\mathrm{mi}}(t_p))\right)^{-1}\mathbf{f}_{\lambda}(\mathbf{H}_{\mathrm{mi}}(t_q)) \\
&= \begin{bmatrix} \mathbf{R}_{\mathrm{mi}}(t_p) & \lambda\mathbf{t}_{\mathrm{mi}}(t_p) \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{R}_{\mathrm{mi}}(t_q) & \lambda\mathbf{t}_{\mathrm{mi}}(t_q) \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{R}_{\mathrm{mi}}(t_p)^{\top} & -\lambda\mathbf{R}_{\mathrm{mi}}(t_p)^{\top}\mathbf{t}_{\mathrm{mi}}(t_p) \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_{\mathrm{mi}}(t_q) & \lambda\mathbf{t}_{\mathrm{mi}}(t_q) \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{R}_{\mathrm{mi}}(t_p)^{\top}\mathbf{R}_{\mathrm{mi}}(t_q) & \lambda\mathbf{R}_{\mathrm{mi}}(t_p)^{\top}\left(\mathbf{t}_{\mathrm{mi}}(t_q) - \mathbf{t}_{\mathrm{mi}}(t_p)\right) \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix} \\
&= \mathbf{f}_{\lambda}(\mathbf{H}_{\mathrm{mi}}^{-1}(t_p)\mathbf{H}_{\mathrm{mi}}(t_q)).
\end{aligned} \tag{2.38}$$

This shows that the knowledge of how the mediary frame aligns with NED is not necessary since this is cancelled out in the calculation of the relative pose. It also shows that the inclusion of the unknown scale parameter does not induce any additional complexity of the problem other than an extra unknown parameter λ.

At this point, naming the relative poses and unknown extrinsic transform as

$$\mathbf{H}_{\mathrm{nb}}^{-1}(t_p)\mathbf{H}_{\mathrm{nb}}(t_q) := \mathbf{A}_{pq} \tag{2.39}$$

$$\mathbf{f}_{\lambda}(\mathbf{H}_{\mathrm{mi}}^{-1}(t_p)\mathbf{H}_{\mathrm{mi}}(t_q)) := \mathbf{B}_{pq}(\lambda) \tag{2.40}$$

$$\mathbf{H}_{\mathrm{bi}} := \mathbf{X} \tag{2.41}$$

the last line of Equation (2.35) becomes

$$\mathbf{A}_{pq}\mathbf{X} = \mathbf{X}\mathbf{B}_{pq}(\lambda). \tag{2.42}$$

This therefore shows how it is possible, with the measurements available on a ship, to formulate a fitting Hand-Eye calibration problem, with the downside that the problem will be *scaleless* and therefore have an extra unknown parameter.

During the derivation, it is required that t_p is a strictly different point in time than t_q, since the case of $t_p = t_q$ causes the relative motion to be the identity, $\mathbf{A} = \mathbf{B} = \mathbf{I}_{4\times 4}$, which in turn results in Equation (2.42) to contain no useful information. It is also clear why the extrinsics as well as the scale factor must be assumed to be constant over the duration of the data to be able to derive the given equality, since otherwise there would be twice as many unknown parameters. The validity of assuming the scale to be constant is highly dubious [22, 25], but alleviated in this project by considering egomotion estimates of short timespans.

The matrix product on each side of Equation (2.42) can be expanded to produce the full set of equations in the *scaleless Hand-Eye equations* seen in Equations (2.44) and (2.45). Note especially that the unknown scale parameter does not appear in Equation (2.44).

$$\begin{bmatrix} \mathbf{R}_{\mathrm{A}} & \mathbf{t}_{\mathrm{A}} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_{\mathrm{X}} & \mathbf{t}_{\mathrm{X}} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R}_{\mathrm{X}} & \mathbf{t}_{\mathrm{X}} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_{\mathrm{B}} & \mathbf{t}_{\mathrm{B}}\lambda \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix} \tag{2.43}$$

$$\Longrightarrow$$

$$\mathbf{R}_{\mathrm{A}}\mathbf{R}_{\mathrm{X}} = \mathbf{R}_{\mathrm{X}}\mathbf{R}_{\mathrm{B}} \tag{2.44}$$

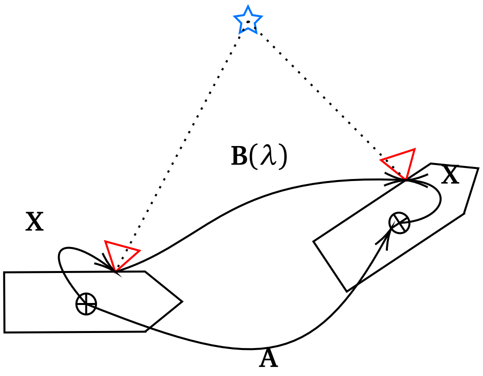
$$\mathbf{R}_{\mathrm{A}}\mathbf{t}_{\mathrm{X}} + \mathbf{t}_{\mathrm{A}} = \mathbf{R}_{\mathrm{X}}\mathbf{t}_{\mathrm{B}}\lambda + \mathbf{t}_{\mathrm{X}} \tag{2.45}$$

A conceptual illustration of the setup is shown in Figure 2.4. It is worth noting the likenesses of this experimental setup to that in Figure 2.3, which further legitimizes the previous derivations.

Multiple methods exist for finding the **X** which solves Equation (2.42) for the unknown extrinsics, either by some closed-form equation or by iterative optimization of a loss function. A summary of select methods is given in Section 2.6.4.

2.6.3 Mathematical properties

What follows is a summary of some of the mathematical properties of Equation (2.42) important for this thesis.



[th]

Figure 2.4: Conceptual illustration of the Hand-Eye Calibration problem for the case of a ship-mounted camera (in red). Note the variable λ representing the unknown scale factor, and how a feature detection algorithm must be employed instead of a calibration plate, with a detected landmark represented as a star.

Relationship of rotation axes

The way rotations of the system are linked through Equation (2.44) has a useful property. Following derivations by Park and Martin in [13] we can rearrange

$$\begin{aligned} \mathbf{R}_\mathrm{A}\mathbf{R}_\mathrm{X} &= \mathbf{R}_\mathrm{X}\mathbf{R}_\mathrm{B} && (2.46) \\ \mathbf{R}_\mathrm{A} &= \mathbf{R}_\mathrm{X}\mathbf{R}_\mathrm{B}\mathbf{R}_\mathrm{X}^\top && (2.47) \end{aligned}$$

and apply the SO(3) logarithm to each side as well as Proposition 1 from Section 2.2.1 we get

$$\begin{aligned} \log(\mathbf{R}_\mathrm{A}) &= \log(\mathbf{R}_\mathrm{X}\mathbf{R}_\mathrm{B}\mathbf{R}_\mathrm{X}^\top) && (2.48) \\ \log(\mathbf{R}_\mathrm{A}) &= \mathbf{R}_\mathrm{X}\log(\mathbf{R}_\mathrm{B})\mathbf{R}_\mathrm{X}^\top. && (2.49) \end{aligned}$$

Further, applying Proposition 2 and the *vee* function gives

$$\begin{aligned} \log(\mathbf{R}_\mathrm{A}) &= (\mathbf{R}_\mathrm{X}\mathrm{Log}(\mathbf{R}_\mathrm{B}))^\wedge && (2.50) \\ \log(\mathbf{R}_\mathrm{A})^\vee = \mathrm{Log}(\mathbf{R}_\mathrm{A}) &= \mathbf{R}_\mathrm{X}\mathrm{Log}(\mathbf{R}_\mathrm{B}). && (2.51) \end{aligned}$$

Equation (2.51) tells us that for any datapair of relative poses $(\mathbf{A}, \mathbf{B})$, the rotation axis of an end-effector movement $\boldsymbol{\alpha} := \mathrm{Log}(\mathbf{R}_\mathrm{A})$ is linked to the rotation axis of a camera movement $\boldsymbol{\beta} := \mathrm{Log}(\mathbf{R}_\mathrm{B})$ through the unknown orientation of the camera relative the end-effector. This property simplifies the coming analysis of observability and will be crucial to describe the properties of data to be used in Hand-Eye calibration.

Observability

Shiu *et al.* showed that to be able to uniquely determine the camera orientation $\mathbf{R}_\mathrm{X}$ which solves the purely rotational Hand-Eye calibration problem given in Equation (2.44), the robot arm must undergo at least two non-zero motions with non-parallel axis of rotation [1]. That is, the extrinsics are observable when for any two datapairs $(\mathbf{R}_{A,1}, \mathbf{R}_{B,1})$ and $(\mathbf{R}_{A,2}, \mathbf{R}_{B,2})$ it holds that

$$\mathrm{Log}(\mathbf{R}_{B,1}) := \boldsymbol{\beta}_1 \nparallel \boldsymbol{\beta}_2 =: \mathrm{Log}(\mathbf{R}_{B,2}). \tag{2.52}$$

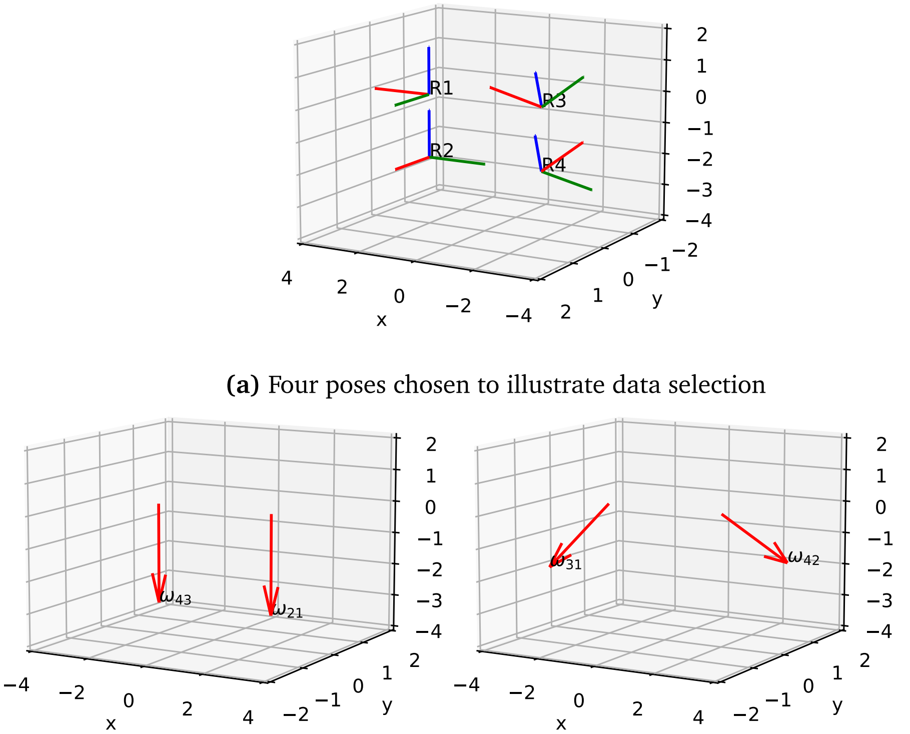
The choice made in this thesis to formulate this requirement in terms of the rotation axes of camera poses as opposed to end-effector poses is arbitrary. This is because the property in Equation (2.51) shows that any geometric analysis of the end-effector rotation axes is equivalent to that of camera rotation axes since the two are related through a constant and structure-preserving rotation $\mathbf{R}_X$. That is; the angle between $\boldsymbol{\beta}_1$ and $\boldsymbol{\beta}_2$ are conserved under this rotation.

Andreff *et al.* additionally showed in [5] how failure to meet the condition in Equation (2.52) results in different indeterminate cases, depending on the nature of the performed motions. Of note for this work is in the case where the system undergoes purely planar motion, that being motion where all translation is contained in a plane and all rotation of the system is performed about the normal to the plane of translation. This means all the rotation axes are parallel. In this case, Andreff *et al.* prove that two nonzero movements still cause the entire extrinsics to be solvable *except* the height of the sensor relative to the plane of motion, but only when the full Hand-Eye equation Equation (2.42) is solved simultaneously. Solving for example for the rotation $\mathbf{R}_X$ alone is not possible. On the upside, their method allows for the estimation of the scale parameter λ under the given conditions.

Construction of relative poses

A simple way to achieve the two relative poses necessary to obtain observability of the extrinsics is to have the robot arm capture four absolute hand- and eye-poses and to compute the relative pose between two and two absolute poses. However, the choice of which absolute poses should be paired to create a single relative pose is not obvious. Consider Figure 2.5. In this figure, four absolute poses of the end-effector have been constructed for illustratory purposes, $\mathbf{H}_1, \mathbf{H}_2, \mathbf{H}_3, \mathbf{H}_4$. When it comes to observability, the relative poses $\mathbf{H}_i^{-1}\mathbf{H}_j$ and $\mathbf{H}_j^{-1}\mathbf{H}_i$ are equivalent, since $\mathrm{Log}(\mathbf{R}_i^\top \mathbf{R}_j) = -\mathrm{Log}(\mathbf{R}_j^\top \mathbf{R}_i)$ [12], thereby offering no new "parallelity". This means that of the four absolute poses, it is possible to generate three unique pairs when permutations of these pairs are considered equal. The rotation axes of two such choices of relative poses calculated are shown in Figures 2.5b and 2.5c, and it is clear that the choice of pairs is not arbitrary. The pair in Figure 2.5b is parallel and therefore does not yield the extrinsics observable, while the pair in Figure 2.5c is seemingly orthogonal.

Tsai *et al.* propose in [6] a methodical way to command the robot arm to produce an optimal set of poses to solve the Hand-Eye calibration problem. When the system cannot be commanded as such, Schmidt *et al.* [7] reformulate the method of Tsai *et al.* to a list of geometric criteria one should try to maximize.



(a) Four poses chosen to illustrate data selection

(b) $\text{Log}(\mathbf{R}_2^\top \mathbf{R}_1)$ and $\text{Log}(\mathbf{R}_4^\top \mathbf{R}_3)$ plotted, the axes are parallel

(c) $\text{Log}(\mathbf{R}_3^\top \mathbf{R}_1)$ and $\text{Log}(\mathbf{R}_4^\top \mathbf{R}_2)$ plotted, the axes are orthogonal

Figure 2.5: Illustration of how the choice of pairing absolute poses to make up a single relative pose matters. Notice how the choice of pairing rotations affects their geometry.

Neither of these are precise formulations on the numerical properties of the input-data, and neither address the challenge posted above of how one should choose pairs of absolute poses to construct a single relative pose. Additionally, despite the theory by Tsai *et al.* giving a quantification of how the geometry of chosen data affects the uncertainty of estimated extrinsics when using their Hand-Eye solver, seen in Equation (2.53), it is not clear how one would go about expanding this theory to concern an arbitrary number of datapoints.

$$\text{Var}(\boldsymbol{\omega}_X) \propto \frac{\sqrt{\text{Var}(\boldsymbol{\beta}_1)^2 + \text{Var}(\boldsymbol{\beta}_2)^2}}{\sin\left[\angle(\boldsymbol{\beta}_1, \boldsymbol{\beta}_2)\right]} \sqrt{\frac{1}{||\boldsymbol{\beta}_1||^2} + \frac{1}{||\boldsymbol{\beta}_2||^2}} \tag{2.53}$$

2.6.4 Hand-Eye solvers

With the Hand-Eye calibration problem formulated and some of its basic mathematical properties quantified, what remains is to actually solve the problem for the unknown extrinsics. In the field of research on the Hand-Eye problem, a multitude of solution techniques, hereby called *solvers*, have been created [2]. These solvers come in many variations based on the chosen parametrization of the poses [24], whether they solve for both the rotation and position of the sensor simultaneously [5] or separately [6, 13] as well as whether the solver uses a closed-form calculation or an iterative procedure [26]. Additionally, most of these solvers naturally give rise to some minimization criteria, which can be applied in any nonlinear least-squares solver.

Previous work of the author in Appendix C has entailed comparing these solvers when the input-data is gathered from ship-sensors. Of the solvers tested, the solution technique of Park and Martin [13] was shown to fare well even when ship-data is considered. In this thesis, the mathematical properties of this solver are explored further, following the results of the previous work.

The *Park-Martin solver* was proposed in their 1994 paper on the topic. Unlike the early papers on Hand-Eye calibration [1, 6], their solver was not derived from geometry but rather from group theory. The Park-Martin solver bases itself on first solving for the camera orientation in Equation (2.51), to then use the estimated orientation when solving for the camera position. They derive the solution to Equation (2.51) to be as shown in Equation (2.54).

$$\begin{aligned} \mathbf{R}_\mathrm{X} &= (\mathbf{M}^\top \mathbf{M})^{-1/2}\mathbf{M}^\top, \text{where} \\ \mathbf{M} &= \sum_{i=1}^{N} \mathrm{Log}\left(\mathbf{R}_{\mathrm{B},i}\right) \mathrm{Log}\left(\mathbf{R}_{\mathrm{A},i}\right)^\top \end{aligned} \tag{2.54}$$

The closed-form solution in Equation (2.54) can alternatively be formulated as a nonlinear optimization problem, which generally is beneficial when the data is corrupted by the presence of noise. In that case, the calibration problem is formulated as a general nonlinear least squares problem, as seen in Equation (2.55), and solved by an iterative solver such as the Levenberg–Marquardt algorithm or BFGS algorithm [20]. Here, f_i is named the *ith residual*, x are the *optimization variables*, $F(x)$ is the cost function and $\rho(\cdot)$ is some weighting function.

$$\min_x F(x) = \frac{1}{2}\sum_i \rho(f_i(x)^2) \tag{2.55}$$

The *Park-Martin residual* to be minimized is then simply the difference between terms in Equation (2.51), that being Equation (2.56).

$$f_i(x) = f_i(\mathbf{R}) = \mathrm{Log}(\mathbf{R}_{\mathrm{A},i}) - \mathbf{R}\mathrm{Log}(\mathbf{R}_{\mathrm{B},i}) \tag{2.56}$$

It should be noted, however, that the solver presented by Park and Martin for estimating the camera position does not take into account the unknown scale parameter present when using egomotion algorithms for constructing the camera movement. Some solvers, like the one presented by Andreff *et al.* in their 2001 paper [5], can additionally estimate the unknown scale. For this project however, the problem of scale is avoided by only performing estimation of the rotational extrinsics.

Chapter 3

Theoretical contributions

Previous work by the author has shown that it is possible to find the orientation of ship-mounted cameras by formulating and solving a fitting Hand-Eye calibration problem. To this end, the Park-Martin Hand-Eye solver was shown to give reasonable results and to be practical, since it is closely tied to the requirements for observability. The Hand-Eye calibration problem is still degenerate in theory for purely planar motions and therefore performance over *nearly* planar ship motions is expected to suffer as a consequence. Further, existing theory on the topic of Hand-Eye calibration is not equipped with the tools necessary to make quantifiable assertions on how the construction, amount and excitation of relative poses affect the estimates. This latter point is especially important when remembering that contrary to the regular Hand-Eye calibration problem for robot-arms, the poses of the system cannot be commanded when using this framework on ship-data, since the ship-movement is already predetermined. The goal of the work in this thesis is to explore these themes and to derive a new theoretical understanding of the Hand-Eye calibration problem and the Park-Martin solver to answer these questions when the input-data is recorded from ship-mounted cameras.

The following chapter presents the theoretical results of this thesis. These results are presented in this chapter to make a clear distinction between previously established theory in literature, given in Chapter 2, and the results of this thesis. Chapter 4 contains simulation results to support these theoretical findings. Unless noted otherwise, the author has not been able to find similar theoretical derivations elsewhere in literature on the same topic.

3.1 Analytic level sets of the Park-Martin cost-function

Existing derivations of properties of the Hand-Eye calibration problem often employ geometric arguments. A consequence of this is that it is difficult to answer questions regarding the numerical properties of the data.

As a first step towards remedying this, this chapter is an exploration of the numerical properties of the Park-Martin optimization residual $\mathbf{r}_i(\mathbf{R}) = \boldsymbol{\alpha}_i - \mathbf{R}\boldsymbol{\beta}_i$ introduced in Section 2.6.4. This is done by analysing the level sets of these residuals, in turn deriving a closed-form parametrization of these sets. The derivation also uses geometric arguments, but the end result is numeric in nature. Analyzing the level sets allows for rephrasing known properties of the Hand-Eye calibration problem in the light of these level sets, as well as building new intuition on the effect of planar data.

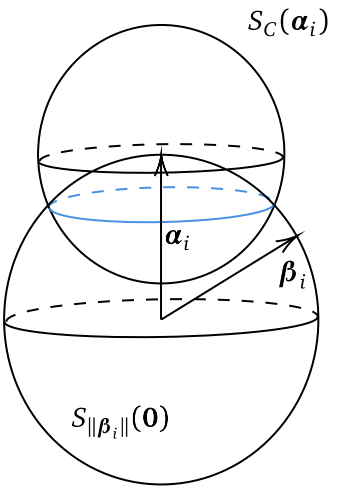

$S_C(\boldsymbol{\alpha}_i)$
$\boldsymbol{\alpha}_i$
$\boldsymbol{\beta}_i$
$S_{||\boldsymbol{\beta}_i||}(\mathbf{0})$

Figure 3.1: This diagram visualizes the points on the intersection of spheres where all rotations $\mathbf{R} \in L_C(||\mathbf{r}_i||)$ must rotate the vector $\boldsymbol{\beta}_i' = \mathbf{R}\boldsymbol{\beta}_i$ onto. Since both $\boldsymbol{\alpha}_i$ and $\boldsymbol{\beta}_i$ are three-dimensional vectors, the spheres are three-dimensional as well. The intersection of the sphere is drawn in blue. The dashed lines are used to aid in showing the 3D shape of the spheres.

3.1.1 Geometric derivation

The level set L_C with level C of a scalar function $f : X \rightarrow Y$ is the subset of the domain for which the function holds the given scalar level, $L_C(f) = \{x \in X \mid f(x) = C\} \subseteq X$ [20].

Solving the Hand-Eye problem can be done by minimizing the cost-function of Park-Martin residuals $F(\mathbf{R}) = \sum_{i=1}^{N} ||\boldsymbol{\alpha}_i - \mathbf{R}\boldsymbol{\beta}_i||_2^2$. We start by analyzing the cost generated by a single datapair and looking at level sets of the normed residual $\mathbf{r}_i' = ||\boldsymbol{\alpha}_i - \mathbf{R}\boldsymbol{\beta}_i|| = C$ at some level C. This is the ith term of the full sum-of-squared-residuals cost-function. Considering $\mathbf{R}\boldsymbol{\beta}_i := \boldsymbol{\beta}_i'$ is a vector, then the set of constant cost C are all rotations that rotate $\boldsymbol{\beta}_i$ onto the sphere with radius C centered at $\boldsymbol{\alpha}_i$, since $S_C(\boldsymbol{\alpha}_i) = \left\{\boldsymbol{\beta}'_i \in \mathbb{R}^3 \text{ such that } ||\boldsymbol{\alpha}_i - \boldsymbol{\beta}'_i|| = C\right\}$ is exactly the definition of this sphere. Rotations additionally preserve the lengths of vectors, and since $\boldsymbol{\alpha}_i = \mathbf{R}\boldsymbol{\beta}_i$, it must hold that $||\boldsymbol{\alpha}_i|| = ||\boldsymbol{\beta}_i||$. Therefore all vectors $\boldsymbol{\beta}'_i$ must also lie on the sphere of preserved length, $S_{||\boldsymbol{\beta}_i||}(\mathbf{0})$, no matter the choice of $\mathbf{R}$. Then the level set $L_C(||\mathbf{r}_i||)$ must be the set of all rotations that move $\boldsymbol{\beta}_i$ onto the intersection of $S_C(\boldsymbol{\alpha}_i)$ and $S_{||\boldsymbol{\beta}_i||}(\mathbf{0})$. This allows for further refinement of the definition of the level set to Equation (3.1).

$$L_C(||\mathbf{r}_i||) = \left\{\mathbf{R} \in \mathrm{SO}(3) \mid \mathbf{R}\boldsymbol{\beta}_i = \boldsymbol{\beta}'_i \in S_C(\boldsymbol{\alpha}_i) \cap S_{||\boldsymbol{\beta}_i||}(\mathbf{0})\right\} \tag{3.1}$$

Figure 3.1 shows an illustration of the developed geometric interpretation thus far. From the figure it is clear that as long as $C < 2||\boldsymbol{\alpha}_i||$, then the intersection $S_C(\boldsymbol{\alpha}_i) \cap S_{||\boldsymbol{\beta}_i||}(\mathbf{0})$ will be a circle. If $C = 2||\boldsymbol{\alpha}_i||$ or $C = 0$, the intersection shrinks to a point antipodal to - or at the tip of - $\boldsymbol{\alpha}_i$, respectively. For higher costs, $C > 2||\boldsymbol{\alpha}_i||$, the spheres do not intersect. Simply put, there exists no rotation which will send $\boldsymbol{\beta}_i$ to the sphere of radius $C > 2||\boldsymbol{\alpha}_i||$, because this would require changing the length of the vector.

All rotations that rotate $\boldsymbol{\beta}_i$ onto the intersection of the spheres may be factorized into two simple rotations: A rotation of $\boldsymbol{\beta}_i$ *onto* the circle of intersection, and a rotation of the resultant vector *around* the circle of intersection. Notably, it is only the former rotation that identifies the level set with any specific value of $(\boldsymbol{\alpha}_i, \boldsymbol{\beta}_i)$, the latter rotation is only an invariant action over the level set of interest. We will name these rotations $\mathbf{R}_1$ and $\mathbf{R}_2$, respectively. There also exists a second invariant action over the level sets, by rotating $\boldsymbol{\beta}_i$ about its own axis. This will not change $\boldsymbol{\beta}_i$, but doing such a rotation before $\mathbf{R}_1$ and $\mathbf{R}_2$ will change the corresponding orientation. Naming this rotation $\mathbf{R}_0$, the level set is completely characterized by the rotations $\mathbf{R} = \mathbf{R}_2\mathbf{R}_1\mathbf{R}_0$, which we now will attempt to assign numerical values.

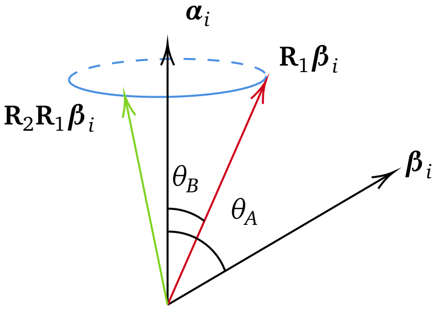

$\boldsymbol{\alpha}_i$
$\mathbf{R}_1\boldsymbol{\beta}_i$
$\mathbf{R}_2\mathbf{R}_1\boldsymbol{\beta}_i$
$\boldsymbol{\beta}_i$
θ_B
θ_A

Figure 3.2: Illustration of the angles and the rotation axes making up $\mathbf{R}_1$ and $\mathbf{R}_2$ in the Park-Martin level sets

3.1.2 Closed-form expression derivation

With this geometric interpretation in hand, we can come to conclusions on the exact numerical values the rotation matrices present in each level set must have. An illustrative figure to aid in the following derivations may be seen in Figure 3.2. The rotation of $\boldsymbol{\beta}_i$ by $\mathbf{R}_1$ is characterized by being about the axis $\boldsymbol{\beta}_i \times \boldsymbol{\alpha}_i$ and some angle θ_1. To find this angle, we define θ_A to be the angle between $\boldsymbol{\alpha}_i$ and $\boldsymbol{\beta}_i$ calculated as $\theta_A = \text{asin}(||\boldsymbol{\beta}_i \times \boldsymbol{\alpha}_i||/(||\boldsymbol{\beta}_i||^2))$, using the previously noted fact that $||\boldsymbol{\alpha}_i|| = ||\boldsymbol{\beta}_i||$. Additionally, we define θ_B to be the angle between $\boldsymbol{\alpha}_i$ and a line segment from the origin to the circle of intersections. By applying the cosine rule and once more the fact that both vector norms are equal, the numerical value of θ_B may be shown to be $\theta_B = \text{acos}((2||\boldsymbol{\beta}_i||^2 - C^2)/(2||\boldsymbol{\beta}_i||^2)))$. Then the rotation angle θ_1 is given by Equation (3.2).

$$\theta_1 = \theta_A - \theta_B = \text{asin}\left(\frac{||\boldsymbol{\beta}_i \times \boldsymbol{\alpha}_i||}{||\boldsymbol{\beta}_i||^2}\right) - \text{acos}\left(\frac{2||\boldsymbol{\beta}_i||^2 - C^2}{2||\boldsymbol{\beta}_i||^2}\right) \tag{3.2}$$

The second rotation $\mathbf{R}_2$ is simply the rotation about the axis $\boldsymbol{\alpha}_i$ with any angle $\theta_2 \in [0, 2\pi]$. It is worth noting how θ_1 is uniquely determinable based on the data $(\boldsymbol{\alpha}_i, \boldsymbol{\beta}_i)$, while the value of θ_2 is to be considered a free variable and therefore a degree of freedom.

Lastly, the rotation of $\boldsymbol{\beta}_i$ about itself, $\mathbf{R}_0$, is also parametrized by the angle-axis formula. This introduces a second free variable $\theta_0 \in [0, 2\pi]$. With this, the Park-Martin residuals' level sets have been fully characterized, and the full expression is given in Equation (3.3).

$$\begin{aligned} L_C(||\mathbf{r}_i||) &= \left\{\mathbf{R} \in \text{SO}(3) \text{ s.t. } ||\boldsymbol{\alpha}_i - \mathbf{R}\boldsymbol{\beta}_i|| = C\right\} \\ &= \left\{\mathbf{R} = \mathbf{R}_2\mathbf{R}_1\mathbf{R}_0 = \mathbf{R}(\theta_2, \boldsymbol{\alpha}_i)\mathbf{R}(\theta_1, \boldsymbol{\beta}_i \times \boldsymbol{\alpha}_i)\mathbf{R}(\theta_0, \boldsymbol{\beta}_i)\right. \\ &\qquad \text{such that } (\theta_0, \theta_2) \in [0, 2\pi]^2 \\ &\qquad \left.\text{and } \theta_1 = \text{asin}\left(\frac{||\boldsymbol{\beta}_i \times \boldsymbol{\alpha}_i||}{||\boldsymbol{\beta}_i||^2}\right) - \text{acos}\left(\frac{2||\boldsymbol{\beta}_i||^2 - C^2}{2||\boldsymbol{\beta}_i||^2}\right)\right\} \end{aligned} \tag{3.3}$$

With the level sets parameterized with the presented closed-form expression, analysis of the level sets can be performed. As a first step, the level sets are plotted to gain some basic understanding. Plotting the rotation matrices themselves is cumbersome and difficult to interpret, so a parametrization must be chosen. For all following analysis, the level sets are analysed and visualized by casting the rotations of the level set onto their Lie algebra equivalents, that being the angle-axis representation in $\mathbb{R}^3$. This allows for plotting the level sets in three dimensions, making analysis much easier. This is achieved by taking the SO(3)-logarithm of each element in $L_C(||\mathbf{r}_i||)$. In this thesis, an implementation of the SO(3)-logarithm with codomain $\boldsymbol{\omega} \in B_\pi(\mathbf{0})$, the ball of radius π, is used.

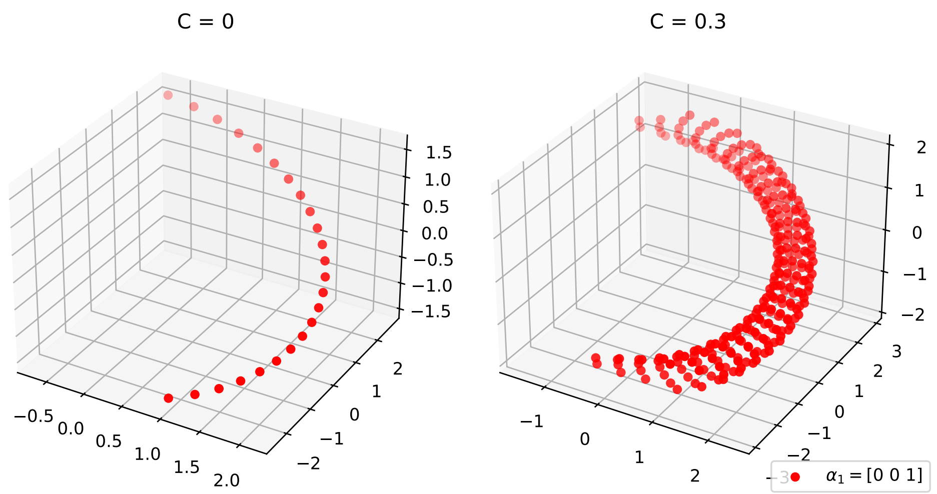


Figure 3.3: Illustration of a single Park-Martin residual level-set, for two different choice of level and arbitrary data

The generated plot using the described method is seen in Figure 3.3 for two different choices of the level C and with a fixed resolution on θ_0 and θ_2. An arbitrary ground-truth extrinsic was chosen to generate the value for $\boldsymbol{\beta}_i$ given the shown values of $\boldsymbol{\alpha}_i$. We will later explore to which degree the extrinsics truly are arbitrary and what the effects are of some other choice of extrinsics, as well as how different choices of the value C affect the expected shape of the residual and cost-functions.

Equation (3.3) will be the basis of further analysis in the following sections in this chapter.

3.1.3 On the effect of different choice of level

From Equation (3.3) one is able to ascertain that the level set is expected to be 2-dimensional, being determined by two free parameters, (θ_0, θ_2). This is expected since rotation matrices are known to have 3 degrees of freedom [12] while enforcing $||\mathbf{r}_i|| = C$ naturally acts as a constraint that removes one of these degrees of freedom. This result is reflected in the experimental results in the rightmost subfigure of Figure 3.3.

As $C \to 0$, the circle of intersection shrinks to a point at the tip of $\boldsymbol{\alpha}_i$. The effect of which is that $\mathbf{R}_1$ simply moves $\boldsymbol{\beta}_i$ to align perfectly with $\boldsymbol{\alpha}_i$, and thereby both rotations $\mathbf{R}_0$ and $\mathbf{R}_2$ will result in the same set of orientations. A sort of gimbal lock occurs, and one degree of freedom is lost which results in one-dimensional level sets. The level set of $C = 0$ is necessarily the level set of which the ground-truth value of $\mathbf{R}_X$ must lie since the positive semi-definiteness of the norm implies $||\boldsymbol{\alpha}_i - \mathbf{R}\boldsymbol{\beta}_i|| = 0 \iff \boldsymbol{\alpha}_i = \mathbf{R}\boldsymbol{\beta}_i$, the latter equation of which $\mathbf{R}_X$ is one of many possible solutions.

This same dimensionality-argument can also be made when $C = 2||\boldsymbol{\alpha}_i||$. When this occurs only a single point of $S_{||\boldsymbol{\beta}_i||}(\mathbf{0})$ may intersect with $S_C(\boldsymbol{\alpha}_i)$, and a similar kind of gimbal lock occurs, only now centered around an orientation different from the ground-truth.

Assume now that we have a single datapoint, $i = 1$ and that the level set $C = 0$ is analyzed. It is clear from the geometric derivation in Section 3.1.1 that any non-zero scaling of $(\boldsymbol{\alpha}_1, \boldsymbol{\beta}_1)$ results in the same level set since the angle between vectors are preserved under scaling. Then conversely if some second datapoint $(\boldsymbol{\alpha}_2, \boldsymbol{\beta}_2)$ is available, as long as this second datapair is non-parallel to the first, the two datapairs' level sets must intersect at a singular point, that point being $\boldsymbol{\omega}_{\mathrm{X}} = \mathrm{Log}(\mathbf{R}_{\mathrm{X}})$. This is because we know from previous analysis in this subsection that the point $\boldsymbol{\omega}_{\mathrm{X}} = \mathrm{Log}(\mathbf{R}_{\mathrm{X}})$ must necessarily lie in both datapairs' $C = 0$ level sets, and that this cannot be true for any other point as this would imply the existence of multiple distinct ground-truth extrinsics. This is assuming that $\mathfrak{so}(3)$ has been bounded to the ball of radius π, as otherwise every rotation vector is congruent with any vector in the same direction with 2π longer length, $\boldsymbol{\omega} \equiv \boldsymbol{\omega} + 2\pi\boldsymbol{\omega}/||\boldsymbol{\omega}||$ under $\mathrm{Exp}(\cdot)$. This analysis is then an alternate proof of the algebraic property shown by other authors [1, 5, 6, 13]: Only two rotations of a non-parallell axis are necessary to – under ideal circumstances – uniquely determine $\mathbf{R}_{\mathrm{X}}$, that being the intersection of the two one-dimensional level curves.

Further, if the data is not perfect, for instance being noisy or uncertain, then their cost evaluated at the ground-truth will not be *exactly* $C = 0$, meaning their level-sets are not one-dimensional. This explains why, through experimental results, one will often be unable to have closed-form solvers converge with any less than 3 datapairs, since any 2 datapairs are not numerically perfect in practice and their level sets therefore do not intersect at a point. This difference between observability *in theory* and observability *in practice* has also been noted by other authors [27].

In Figure 3.4, two one-dimensional level sets of non-parallel datapairs are plotted along with the ground truth extrinsics. By banal example, it is clear that the $C = 0$ level sets intersect in a single point, that point being the ground-truth extrinsics.

3.1.4 On the effect of summation

Formulating a closed-form expression of the full cost-function's level sets using the level sets of the residuals is not easy, and does not lend itself to easy analysis. Letting $F(\mathbf{R}) = \frac{1}{2}\sum_{i=1}^{N} ||\mathbf{r}_i||^2$ be the cost-function, an expression for the level set of level C of the cost-function $F(\mathbf{R})$ is seen in Equation (3.4). The summing of all single residuals' contribution on the full cost allows for some residuals to attain a lower value than the average of $\overline{C} = \sqrt{\frac{2C}{N}}$ required, if the other residuals evaluated at the same rotation attains a correspondingly higher value. The level set of the full cost-function is therefore the orientations in the intersection of all residuals' level sets, with some leeway loosely speaking. The summing of individual residuals also allows for the presence of noise, which corresponds to the costs be-

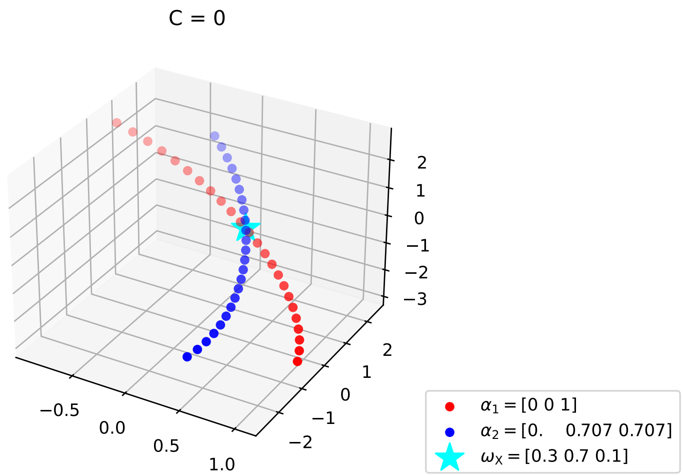


Figure 3.4: The $C = 0$ Park-Martin level sets are drawn for the two given values for $\boldsymbol{\alpha}_i$ and for an arbitrary choice of ground-truth extrinsics. The ground-truth's angle-axis parameters is marked in cyan. Notice the two level sets intersecting in the cyan point.

ing slightly larger or smaller than they would be if the cost of the noiseless data was evaluated at the same orientation.

$$L_C(F) = \left\{ \mathbf{R} \in \bigcap_{i=1}^{N} L_{C_i}(||\mathbf{r}_i||) \text{ such that } \frac{1}{2}\sum_{i=1}^{N} C_i^2 = C \right\} \tag{3.4}$$

3.1.5 On the relationship between multiple data-pairs

In the previous subsections, the level sets of the Park-Martin residual have been plotted by arbitrarily selecting $\boldsymbol{\alpha} = [0, 0, 1]^\top$ and calculating the value for $\boldsymbol{\beta}$ given some, also arbitrarily chosen, ground-truth extrinsics $\mathbf{R}_\mathrm{X}$. This has been done for simplicity in illustration since the ship-data for which this project is focused on will have data mostly pointing along the Z-axis, reflecting its planar nature. It is of interest to note to which degree these choices indeed are arbitrary, or if the choice of data and extrinsics will affect the properties of these level sets. The former is analyzed in this section.

Let $(\boldsymbol{\alpha}_i, \boldsymbol{\beta}_i)$ be a datapair for which the level sets already have been determined. Any second datapair $(\tilde{\boldsymbol{\alpha}}_i, \tilde{\boldsymbol{\beta}}_i)$ can be related to the first through some rotation $\tilde{\mathbf{R}}$, through the formulas $\tilde{\boldsymbol{\alpha}}_i = \tilde{\mathbf{R}}\boldsymbol{\alpha}_i$ and $\tilde{\boldsymbol{\beta}}_i = \tilde{\mathbf{R}}\boldsymbol{\beta}_i$. The rotation $\tilde{\mathbf{R}}$ is decomposed as two rotations, first a rotation aligning $\boldsymbol{\beta}_i$ and $\tilde{\boldsymbol{\beta}}_i$ and a second rotation about $\tilde{\boldsymbol{\beta}}_i$ which then aligns $\boldsymbol{\alpha}_i$ and $\tilde{\boldsymbol{\alpha}}_i$ without changing $\tilde{\boldsymbol{\beta}}_i$. Then the following derivation shows the relationship between the datapairs' level sets.

We begin by defining the level set of $(\boldsymbol{\alpha}_i, \boldsymbol{\beta}_i)$ and performing the non-destructive action of multiplying with the identity:

$$\begin{aligned}||\boldsymbol{\alpha}_i - \mathbf{R}\boldsymbol{\beta}_i|| &= C\\ ||\mathbf{I}_{3\times3}\boldsymbol{\alpha}_i - \mathbf{R}\mathbf{I}_{3\times3}\boldsymbol{\beta}_i|| &= C\\ ||\tilde{\mathbf{R}}^\top\tilde{\mathbf{R}}\boldsymbol{\alpha}_i - \mathbf{R}\tilde{\mathbf{R}}^\top\tilde{\mathbf{R}}\boldsymbol{\beta}_i|| &= C.\end{aligned} \tag{3.5}$$

We recognize $\tilde{\boldsymbol{\alpha}}_i$ and $\boldsymbol{\beta}'_i$ in the last line,

$$||\tilde{\mathbf{R}}^\top\tilde{\boldsymbol{\alpha}}_i - \mathbf{R}\tilde{\mathbf{R}}^\top\tilde{\boldsymbol{\beta}}_i|| = C. \tag{3.6}$$

We know that performing a rotation of a vector will not change the norm of the product, by definition of a rotation, so we are free to multiply the term inside the norm with a rotation matrix of our choice. In this case, we multiply with $\tilde{\mathbf{R}}$:

$$\begin{aligned}||\tilde{\mathbf{R}}^\top\tilde{\boldsymbol{\alpha}}_i - \mathbf{R}\tilde{\mathbf{R}}^\top\tilde{\boldsymbol{\beta}}_i|| = ||\tilde{\mathbf{R}}(\tilde{\mathbf{R}}^\top\tilde{\boldsymbol{\alpha}}_i - \mathbf{R}\tilde{\mathbf{R}}^\top\tilde{\boldsymbol{\beta}}_i)|| &= C\\ ||\tilde{\mathbf{R}}\tilde{\mathbf{R}}^\top\tilde{\boldsymbol{\alpha}}_i - \tilde{\mathbf{R}}\mathbf{R}\tilde{\mathbf{R}}^\top\tilde{\boldsymbol{\beta}}_i|| &= C\\ ||\tilde{\boldsymbol{\alpha}}_i - (\tilde{\mathbf{R}}\mathbf{R}\tilde{\mathbf{R}}^\top)\tilde{\boldsymbol{\beta}}_i|| &= C.\end{aligned} \tag{3.7}$$

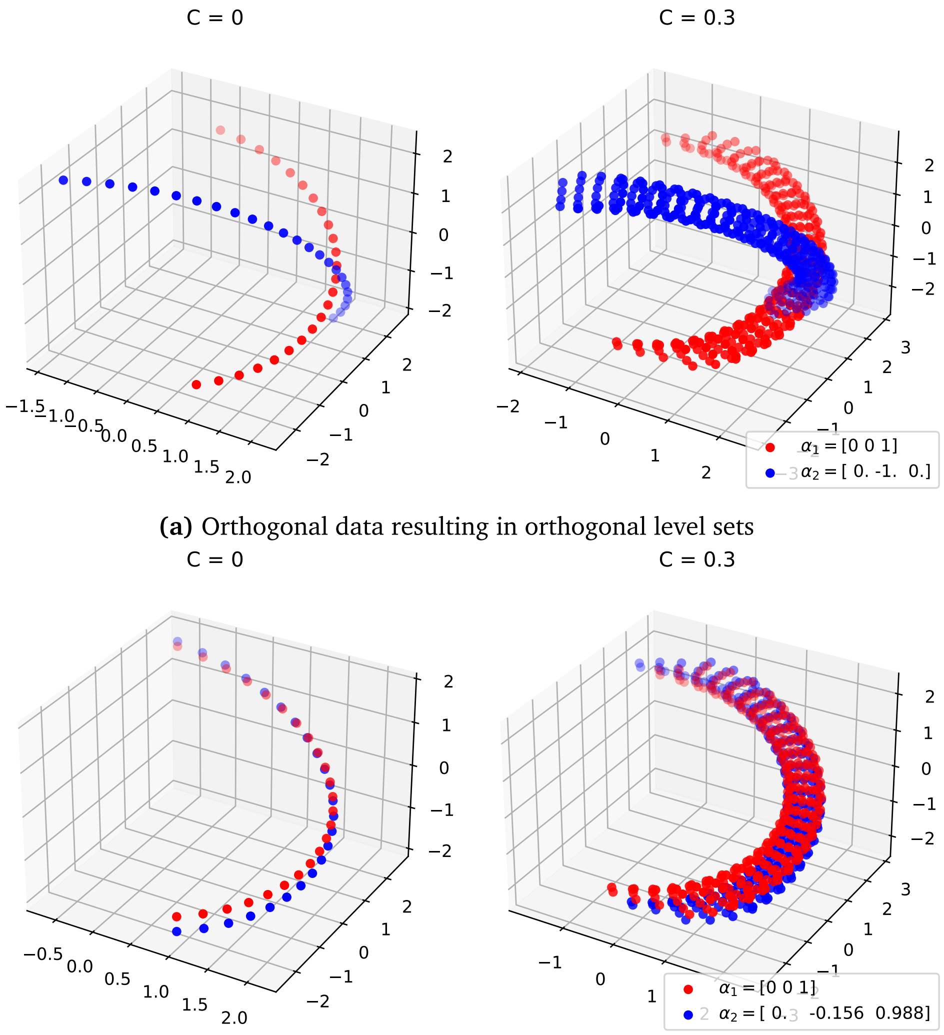
At this point, we can see that the level sets of this new datapair has elements $\mathbf{R}_{\text{new}} := \tilde{\mathbf{R}}\mathbf{R}\tilde{\mathbf{R}}^\top$. In this work, we opted for parameterizing the rotations of each level set in the Lie algebra. Two properties of the SO(3) Lie algebra will prove useful further, namely the property that $\log\left(\mathbf{X}\mathbf{B}\mathbf{X}^\top\right) = \mathbf{X}\log(\mathbf{B})\mathbf{X}^\top$ and $\mathbf{X}\log(\mathbf{B})\mathbf{X}^\top = (\mathbf{X}\mathrm{Log}(\mathbf{B}))^\wedge$. Applying this property on our new level set's elements we see that

$$\begin{aligned}\log(\mathbf{R}_{\text{new}}) = \log(\tilde{\mathbf{R}}\mathbf{R}\tilde{\mathbf{R}}^\top) = \tilde{\mathbf{R}}\log(\mathbf{R})\tilde{\mathbf{R}}^\top = \left(\tilde{\mathbf{R}}\mathrm{Log}(\mathbf{R})\right)^\wedge\\ \implies \mathrm{Log}(\mathbf{R}_{\text{new}}) = \tilde{\mathbf{R}}\mathrm{Log}(\mathbf{R})\end{aligned} \tag{3.8}$$

This result shows that if the level set is parameterized over the Lie algebra, then we expect the level set of any new datapoints to be rotated equally to how the data is rotated.

The conclusion is validated through simulations. In Section 3.1.5 we see how the different choice of the second datapoint affects the level set relative to the first datapoint. Orthogonal choice of data results in orthogonal level sets, and nearly parallel data results in nearly parallel level sets.

These results in turn hint towards a connection between the planarity of ship-movements and differing uncertainty in differing directions of the search space. Figure 3.5b illustrates that the level set of any nonzero cost will be much less uncertain in the direction orthogonal to the level sets, where these do not overlap and therefore have an empty intersection, as opposed to directions tangential to the level sets, where many different rotations give (nearly) the same cost. This is not as much a case in Figure 3.5a, where the overlap between the sets is minimal. This fact is compounded by the fact that the direction of largest descent, that being all directions orthogonal to the level sets, roughly coincide in a planar dataset, while the same is not true for a dataset with a more diverse set of rotation vectors present.



(a) Orthogonal data resulting in orthogonal level sets

(b) Nearly parallel data resulting in nearly parallel level sets

Figure 3.5: Illustrative figure of how rotating the data likewise rotates the level sets, as shown with orthogonal and planar data. The same ground truth extrinsics were used for both of the two cases.

3.1.6 On the effect of different ground-truth extrinsics

The analysis so far has employed arbitrary choices for the ground-truth extrinsics, with the hope that making some different choice of extrinsics would not have any adverse effects on the properties of these level sets. That is, the properties derived of the level set with some ground-truth, $\mathbf{R}_\mathrm{X}$, will be the same as the properties as the level set with different ground-truth, $\tilde{\mathbf{R}}\mathbf{R}_\mathrm{X}$. This is explicitly checked in this subsection.

Let $L_C(||\mathbf{r}_i||)$ be the level set of the residual associated with datapair $(\boldsymbol{\alpha}_i, \boldsymbol{\beta}_i)$, where $\boldsymbol{\alpha}_i = \mathbf{R}_\mathrm{X}\boldsymbol{\beta}_i$. Further we define $L_C(||\mathbf{s}_i||)$ to be the level set of the residual with datapair $(\tilde{\boldsymbol{\alpha}}_i, \boldsymbol{\beta}_i)$, where $\tilde{\boldsymbol{\alpha}}_i = \tilde{\mathbf{R}}\mathbf{R}_\mathrm{X}\boldsymbol{\beta}_i$. This second datapair thereby has entirely separate, but relatable, ground-truth extrinsics than the first, $\tilde{\mathbf{R}}\mathbf{R}_\mathrm{X}$. Choosing to let both datapairs share the camera rotation vector $\boldsymbol{\beta}_i$ can be done without loss of generality, see discussion in Section 3.1.5. From these definitions, we can derive that

$$\begin{aligned}
&\tilde{\boldsymbol{\alpha}}_i = \tilde{\mathbf{R}}\mathbf{R}_\mathrm{X}\boldsymbol{\beta}_i \\
&\implies \mathbf{R}_\mathrm{X}^\top\tilde{\mathbf{R}}^\top\tilde{\boldsymbol{\alpha}}_i = \boldsymbol{\beta}_i \\
&\implies \boldsymbol{\alpha}_i = \mathbf{R}_\mathrm{X}\boldsymbol{\beta}_i = \mathbf{R}_\mathrm{X}\mathbf{R}_\mathrm{X}^\top\tilde{\mathbf{R}}^\top\tilde{\boldsymbol{\alpha}}_i \\
&\implies \tilde{\boldsymbol{\alpha}}_i = \tilde{\mathbf{R}}\boldsymbol{\alpha}_i.
\end{aligned} \tag{3.9}$$

This means that if the former residual has elements of its level set parameterized by $\mathbf{R} = \mathbf{R}_2(\theta_2, \boldsymbol{\alpha}_i)\mathbf{R}_1(\angle(\boldsymbol{\beta}_i, \boldsymbol{\alpha}_i) - \theta_B, \boldsymbol{\beta}_i \times \boldsymbol{\alpha}_i)\mathbf{R}_0(\theta_0, \boldsymbol{\beta}_i)$, then the latter residual will have level set with elements $\mathbf{R} = \mathbf{R}_2(\theta_2, \tilde{\mathbf{R}}\boldsymbol{\alpha}_i)\mathbf{R}_1((\angle(\boldsymbol{\beta}_i, \tilde{\mathbf{R}}\boldsymbol{\alpha}_i) - \theta_B, \boldsymbol{\beta}_i \times \tilde{\mathbf{R}}\boldsymbol{\alpha}_i)\mathbf{R}_0(\theta_0, \boldsymbol{\beta}_i)$.

We can see that the only effect differing extrinsics will have on the level sets is that the axis about which the second and third rotations are performed have been slightly modified. How this affects the corresponding angle-axis parametrization and thereby the plots from before is more difficult to say. Arguments about the sets' dimensionality, as related through the relevant level, will still hold even if the axes of rotation change.

3.1.7 Boundedness of optimization

The geometric derivation of the level sets performed in Section 3.1.1 shows how the cost of any single residual is bounded by above by $||\mathbf{r}_i(\mathbf{R})|| \leq 2||\boldsymbol{\alpha}_i||$, for any choice of rotation $\mathbf{R}$. This in turn implies the existence of an absolute upper bound on the cost-function, that being

$$F(\mathbf{R}) = \frac{1}{2}\sum_{i=1}^{N} ||\mathbf{r}_i||^2 \leq \frac{1}{2}\sum_{i=1}^{N}(2||\boldsymbol{\alpha}_i||)^2 = \sum_{i=1}^{N} 2||\boldsymbol{\alpha}_i||^2 := F(\tilde{\mathbf{R}}). \tag{3.10}$$

The cost-function takes on the value of this upper bound in a dataset where all $\boldsymbol{\alpha}_i$ are parallel, no noise is present and the cost function is evaluated at any rotation

$\tilde{\mathbf{R}}$ sending $\boldsymbol{\beta}_i$ to the point antipodal of $\boldsymbol{\alpha}_i$. The set of rotations achieving this can be expressed as

$$\tilde{\mathbf{R}} = \mathbf{R}(\theta_2, \boldsymbol{\alpha}_i)\mathbf{R}(\theta_1 - \pi, \boldsymbol{\beta}_i \times \boldsymbol{\alpha}_i)\mathbf{R}(\theta_0, \boldsymbol{\beta}_i). \tag{3.11}$$

The Hand-Eye calibration problem with Park-Martin cost is then a bounded optimization problem over the compact domain SO(3). This should not come as a surprise, since the cost function is continuous over a compact domain, thereby fulfilling the extreme value theorem's criteria. A numerical value for this upper bound, however, is enabled by the presented geometric derivation. This bound can be useful when performing iterative nonlinear optimization of the cost-function, but care must be taken. This is because when such optimization procedures are performed it is most often performed over a parametrization of SO(3). This is also done in this thesis, using the exponential map to map rotation vectors into rotation matrices. The exponential map is known to be surjective, but not injective, and in fact it is periodic in the input angle, see Equation (2.12) in Section 2.2.1. This implies the cost-function is periodic as well as non-convex when viewed over the entirety of $\mathbb{R}^3$. This has the disadvantage that if the domain of the parameterized cost-function is enforced to be closed, with the intent of exploiting the derived upper bound, then the iterative optimization may wrongfully converge to a local minima at the boundary. See Figure 3.6 for an illustration of this. For this reason, optimization performed in this work is not performed over a closed subset of $\mathbb{R}^3$, and the upper bound derived above is not exploited further.

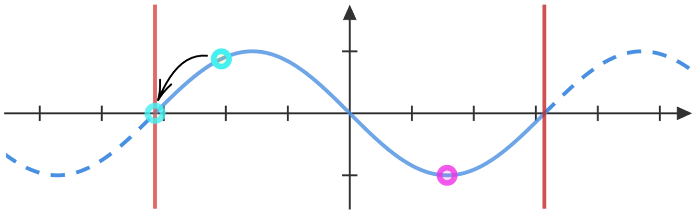


Figure 3.6: Illustration of an iterative optimization of a periodic cost function over a closed domain, where the iterative scheme wrongfully converges to a local minima at the boundary. The optimum is marked in magenta, the steps of the iterative solver in cyan and the upper- and lower boundaries of the domain are shown in red.

3.2 Noise-propagation of the Park-Martin residuals

Recent advances in the fields of estimation and navigation have relied more and more on stochastic and nonlinear models, as opposed to deterministic linear models [19, 21, 28]. With these developments as motivation, it seems worthwhile to investigate how probabilistic modelling can be used for the Hand-Eye problem as well. This will be done in this thesis by deriving the covariance of the Park-Martin residuals $\mathbf{r}_i(\mathbf{R}) = \boldsymbol{\alpha}_i - \mathbf{R}\boldsymbol{\beta}_i$ given a model of the noise on both ship- and camera-orientations. A challenge lies in these measurements not being vectors, but rather poses, and how to model noise over such measurements. Additionally, the poses themselves are not input to the Park-Martin residuals, but rather the relative pose between two absolute measurements. This will lead to further complication.

3.2.1 Naïve approach

We begin the derivation by assuming that the noise over each datapair $(\boldsymbol{\alpha}_i, \boldsymbol{\beta}_i)$ is known and given by the simple measurement noise model presented in Section 2.2.2. Let $\boldsymbol{\alpha}_i = \bar{\boldsymbol{\alpha}}_i + \mathbf{z}_{\alpha,i}$, with $\bar{\boldsymbol{\alpha}}_i$ being the "true" relative body orientation number i and $\mathbf{z}_{\alpha,i} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\alpha,i})$. This in turn means that $\boldsymbol{\alpha}_i \sim \mathcal{N}(\bar{\boldsymbol{\alpha}}_i, \boldsymbol{\Sigma}_{\alpha,i})$. Similarly let $\boldsymbol{\beta}_i = \bar{\boldsymbol{\beta}}_i + \mathbf{z}_{\beta,i}$ with $\mathbf{z}_{\beta,i} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\beta,i})$.

Inserting these into the Park-Martin residuals gives

$$
\begin{aligned}
\mathbf{r}_i(\mathbf{R}) &= \boldsymbol{\alpha}_i - \mathbf{R}\boldsymbol{\beta}_i && (3.12)\\
&= \bar{\boldsymbol{\alpha}}_i + \mathbf{z}_{\alpha,i} - \mathbf{R}(\bar{\boldsymbol{\beta}}_i + \mathbf{z}_{\beta,i}) && (3.13)\\
&= \bar{\boldsymbol{\alpha}}_i - \mathbf{R}\bar{\boldsymbol{\beta}}_i + \mathbf{z}_{\alpha,i} - \mathbf{R}\mathbf{z}_{\beta,i}. && (3.14)
\end{aligned}
$$

The ith Park-Martin residual is simply a linear transformation of the stochastic measurements, which means the propagation of uncertainty is simple [19]. The residual is then distributed by $\mathbf{r}_i(\mathbf{R}) \sim \mathcal{N}(\bar{\boldsymbol{\alpha}}_i - \mathbf{R}\bar{\boldsymbol{\beta}}_i,\ \boldsymbol{\Sigma}_{\alpha,i} + \mathbf{R}\boldsymbol{\Sigma}_{\beta,i}\mathbf{R}^\top)$. Note also that the noiseless measurements still must fulfill $\bar{\boldsymbol{\alpha}}_i = \mathbf{R}_\mathrm{X}\bar{\boldsymbol{\beta}}_i$, meaning the residual evaluated at the ground truth orientation will have expression $\mathbf{r}_i(\mathbf{R}_\mathrm{X}) = \mathbf{z}_{\alpha,i} - \mathbf{R}_\mathrm{X}\mathbf{z}_{\beta,i}$, and thereby must be distributed by $\mathbf{r}_i(\mathbf{R}_\mathrm{X}) \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\alpha,i} + \mathbf{R}_\mathrm{X}\boldsymbol{\Sigma}_{\beta,i}\mathbf{R}_\mathrm{X}^\top)$.

Defining the covariance of the residual to be

$$
\boldsymbol{\Sigma}_{r,i} := \boldsymbol{\Sigma}_{\alpha,i} + \mathbf{R}\boldsymbol{\Sigma}_{\beta,i}\mathbf{R}^\top, \tag{3.15}
$$

then this covariance may be used when minimizing the Park-Martin residuals using the Mahalonobis norm. If an estimate of the body- and sensor-covariance for each point in time is known then we can compensate for the uncertainty of datapair i by scaling the residual by the inverse of the derived covariance of the datapair's residual, thus achieving a *whitening* of the residuals' distribution. This is done as

$$
F(\mathbf{R}) = \frac{1}{2}\sum_{i=1}^{N} \mathbf{r}_i^\top \boldsymbol{\Sigma}_{r,i}^{-1} \mathbf{r}_i = \frac{1}{2}\sum_{i=1}^{N} ||\boldsymbol{\Sigma}_{r,i}^{-\frac{1}{2}} \mathbf{r}_i||^2. \tag{3.16}
$$

This should in theory allow residuals with low uncertainty to be weighted more heavily than the residuals with high uncertainty, and thus better estimates should be achieved.

A vital assumption for this to work is that an accurate estimate of the residuals' covariance is being used. Note for example how the covariance of the residuals is a function of the parameter to be estimated, **R**. This can potentially lead to challenges when far away from the ground-truth value.

We denote the residual covariance in Equation (3.15) as a *naïve* covariance, as it assumes the covariance of the relative rotations $(\boldsymbol{\alpha}_i, \boldsymbol{\beta}_i)$ is known. This will not necessarily be the case for real-world applications, and we will see how we deal with this further.

3.2.2 Group-theoretic covariance

With Equation (3.15) we find how covariance compensation can be performed when the covariance on the relative rotation vectors $(\boldsymbol{\alpha}_i, \boldsymbol{\beta}_i)$ is known. In a real-world scenario, however, it is very possible that one only knows the uncertainty of the absolute orientation measurements making up these relative rotation vectors. Luckily, using recent developments within robotics on representing and propagating noise over group-elements, we will still be able to derive an expression of the corresponding covariance of $(\boldsymbol{\alpha}_i, \boldsymbol{\beta}_i)$.

To this end, let $\mathbf{R}_{\mathrm{nb}}(t_j) := \mathbf{R}_{\mathrm{n},j}$ and $\mathbf{R}_{\mathrm{nb}}(t_k) := \mathbf{R}_{\mathrm{n},k}$ be the absolute measured orientation at timestamps $t_j,\ t_k$. Recall from Section 2.2.2 that we model these measurements as noisy by either right or left exponentiating them with a vectorial noise.

Following the method of Mangelson *et al.* [14] we first examine the left exponentiation expression, meaning the noise vector and its covariance is defined relative to the identity element. This is to be understood as the coordinates of the noise and covariance being given in the world frame, n. Written up, we model the measurements as

$$\begin{aligned} \mathbf{R}_{\mathrm{n},j} &= \mathrm{Exp}(\boldsymbol{\omega}_{\mathrm{n},j}) \circ \bar{\mathbf{R}}_{\mathrm{n},j}, \quad \boldsymbol{\omega}_{\mathrm{n},j} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\mathrm{n},j}) \\ \mathbf{R}_{\mathrm{n},k} &= \mathrm{Exp}(\boldsymbol{\omega}_{\mathrm{n},k}) \circ \bar{\mathbf{R}}_{\mathrm{n},k}, \quad \boldsymbol{\omega}_{\mathrm{n},k} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\mathrm{n},k}). \end{aligned} \tag{3.17}$$

Mangelson *et al.* then show how the uncertainty propagates through the operation of taking the relative rotation $\mathbf{R}_{jk} := \mathbf{R}_{\mathrm{n},j}^{\top}\mathbf{R}_{\mathrm{n},k}$ will lead to it also being a randomly distributed rotation, distributed by

$$\mathbf{R}_{jk} = \mathrm{Exp}(\boldsymbol{\omega}_{jk}) \circ \bar{\mathbf{R}}_{jk}, \quad \boldsymbol{\omega}_{jk} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{jk}) \tag{3.18}$$

where

$$\bar{\mathbf{R}}_{jk} = \bar{\mathbf{R}}_{\mathrm{n},j}^{\top} \circ \bar{\mathbf{R}}_{\mathrm{n},k}, \tag{3.19}$$

and the covariance is as seen in Equation (3.20).

$$\begin{aligned}\boldsymbol{\Sigma}_{jk} \approx &\mathbf{Ad}_{\bar{\mathbf{R}}_{\mathrm{n},j}^{-1}}\boldsymbol{\Sigma}_{\mathrm{n},j}\mathbf{Ad}_{\bar{\mathbf{R}}_{\mathrm{n},j}^{-1}}^{\top}\\ &+\mathbf{Ad}_{\bar{\mathbf{R}}_{\mathrm{n},j}^{-1}}\boldsymbol{\Sigma}_{\mathrm{n},k}\mathbf{Ad}_{\bar{\mathbf{R}}_{\mathrm{n},j}^{-1}}^{\top}\\ &-\mathbf{Ad}_{\bar{\mathbf{R}}_{\mathrm{n},j}^{-1}}\boldsymbol{\Sigma}_{\mathrm{n}j,\mathrm{n}k}\mathbf{Ad}_{\bar{\mathbf{R}}_{\mathrm{n},j}^{-1}}^{\top}\\ &-\mathbf{Ad}_{\bar{\mathbf{R}}_{\mathrm{n},j}^{-1}}\boldsymbol{\Sigma}_{\mathrm{n}j,\mathrm{n}k}^{\top}\mathbf{Ad}_{\bar{\mathbf{R}}_{\mathrm{n},j}^{-1}}^{\top}\end{aligned} \tag{3.20}$$

Using the definition of the adjoint over SO(3) and some simplifications, Equation (3.20) simplifies to Equation (3.21).

$$\boldsymbol{\Sigma}_{jk} \approx \bar{\mathbf{R}}_{\mathrm{n},j}^{\top}\left(\boldsymbol{\Sigma}_{\mathrm{n},j} + \boldsymbol{\Sigma}_{\mathrm{n},k} - 2\boldsymbol{\Sigma}_{\mathrm{n}j,\mathrm{n}k}\right)\bar{\mathbf{R}}_{\mathrm{n},j} \tag{3.21}$$

Alternatively, if the noise and covariances are modelled as being defined locally, that is; about the current orientation, then [25] shows how the covariance and mean rotation of the relative rotation can be developed similarly. For this, the noise enters the rotation through the right exponentiation. We define the noisy rotations this time as

$$\begin{aligned}\mathbf{R}_{\mathrm{n},j} = \bar{\mathbf{R}}_{\mathrm{n},j} \circ \mathrm{Exp}(\boldsymbol{\omega}_{\mathrm{n},j}), \quad \boldsymbol{\omega}_{\mathrm{n},j} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\mathrm{n},j})\\ \mathbf{R}_{\mathrm{n},k} = \bar{\mathbf{R}}_{\mathrm{n},k} \circ \mathrm{Exp}(\boldsymbol{\omega}_{\mathrm{n},k}), \quad \boldsymbol{\omega}_{\mathrm{n},k} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\mathrm{n},k}),\end{aligned} \tag{3.22}$$

and once more inspect the noise on the relative rotation $\mathbf{R}_{jk} := \mathbf{R}_{\mathrm{n},j}^{\top}\mathbf{R}_{\mathrm{n},k}$, finding it is distributed by

$$\mathbf{R}_{jk} = \bar{\mathbf{R}}_{jk} \circ \mathrm{Exp}(\boldsymbol{\omega}_{jk}), \quad \boldsymbol{\omega}_{jk} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{jk}), \tag{3.23}$$

where the covariance of relative rotation noise now admits the form in Equation (3.24).

$$\boldsymbol{\Sigma}_{jk} \approx \bar{\mathbf{R}}_{jk}^{\top}\boldsymbol{\Sigma}_{\mathrm{n},j}\bar{\mathbf{R}}_{jk} + \boldsymbol{\Sigma}_{\mathrm{n},k} - 2\bar{\mathbf{R}}_{jk}^{\top}\boldsymbol{\Sigma}_{\mathrm{n}j,\mathrm{n}k} \tag{3.24}$$

No matter which of the two models is being used, the last step in our derivation is the same. We have shown how the relative rotation $\mathbf{R}_{jk}$ is perturbed by the vectorial noise vector $\boldsymbol{\omega}_{jk}$, as well as derived the distribution of the latter. The object of interest for the Park-Martin residual, however, is the rotation vector of the relative pose, $\mathrm{Log}(\mathbf{R}_{jk})$. Taking this logarithm, and using the Baker–Campbell–Hausdorf (BCH) approximation given in Section 2.2.4, the rather practical form in Equation (3.25) is found.

$$\begin{gathered}\mathrm{Log}(\mathrm{Exp}(\boldsymbol{\omega}_{jk}) \circ \bar{\mathbf{R}}_{jk}) \approx \mathrm{Log}(\mathrm{Exp}(\boldsymbol{\omega}_{jk})) + \mathrm{Log}(\bar{\mathbf{R}}_{jk}) = \boldsymbol{\omega}_{jk} + \boldsymbol{\alpha}_{jk}\\ \text{or equally, if right-exponentiation is used:}\\ \mathrm{Log}(\bar{\mathbf{R}}_{jk} \circ \mathrm{Exp}(\boldsymbol{\omega}_{jk})) \approx \mathrm{Log}(\bar{\mathbf{R}}_{jk}) + \mathrm{Log}(\mathrm{Exp}(\boldsymbol{\omega}_{jk})) = \boldsymbol{\alpha}_{jk} + \boldsymbol{\omega}_{jk}\end{gathered} \tag{3.25}$$

This is exactly the form assumed in Equation (3.13) for the naïve covariance compensation. This is of course an approximation, but since the noise should often be a small value relative the orientation then we can hope the approximation error to be small as well.

The derivation above is equal for the camera-rotations, and as such the covariance of both $\boldsymbol{\alpha}_i$ and $\boldsymbol{\beta}_i$ have been found as expressions of the covariances on the absolute measurements. Given these derivations and combining them with Equation (3.15), we get the actual covariance of Park-Martin residual i given covariances on absolute rotation measurements in Equations (3.26) and (3.27). Please note the newly introduced symbol $\bar{\mathbf{G}}_{jk} := \bar{\mathbf{G}}_{\mathrm{m},j}^\top \bar{\mathbf{G}}_{\mathrm{m},k}$ as the relative *camera* rotations, and distinguish this from the previously defined relative *ship* rotations $\bar{\mathbf{R}}_{jk}$.

$$\begin{aligned} \boldsymbol{\Sigma}_{r,i} &= \boldsymbol{\Sigma}_{\alpha,i} + \mathbf{R}\boldsymbol{\Sigma}_{\beta,i}\mathbf{R}^\top \\ &\approx \bar{\mathbf{R}}_{\mathrm{n},j}^\top \left(\boldsymbol{\Sigma}_{\mathrm{n},j} + \boldsymbol{\Sigma}_{\mathrm{n},k} - 2\boldsymbol{\Sigma}_{\mathrm{n}j,\mathrm{n}k}\right)\bar{\mathbf{R}}_{\mathrm{n},j} + \mathbf{R}\bar{\mathbf{R}}_{\mathrm{m},j}^\top \left(\boldsymbol{\Sigma}_{\mathrm{m},j} + \boldsymbol{\Sigma}_{\mathrm{m},k} - 2\boldsymbol{\Sigma}_{\mathrm{m}j,\mathrm{m}k}\right)\bar{\mathbf{R}}_{\mathrm{m},j}\mathbf{R}^\top \end{aligned} \tag{3.26}$$

or for the case of local covariances

$$\boldsymbol{\Sigma}_{r,i} \approx \bar{\mathbf{R}}_{jk}^\top \boldsymbol{\Sigma}_{\mathrm{n},j}\bar{\mathbf{R}}_{jk} + \boldsymbol{\Sigma}_{\mathrm{n},k} - 2\bar{\mathbf{R}}_{jk}^\top \boldsymbol{\Sigma}_{\mathrm{n}j,\mathrm{n}k} + \mathbf{R}^\top \left(\bar{\mathbf{G}}_{jk}^\top \boldsymbol{\Sigma}_{\mathrm{n},j}\bar{\mathbf{G}}_{jk} + \boldsymbol{\Sigma}_{\mathrm{n},k} - 2\bar{\mathbf{G}}_{jk}^\top \boldsymbol{\Sigma}_{\mathrm{n}j,\mathrm{n}k}\right)\mathbf{R}^\top \tag{3.27}$$

3.2.3 Some considerations and challenges

The developed covariance of the Park-Martin residual makes use of several assumptions and simplifications, some of which may impact the actual performance of using this for covariance compensation.

Firstly, Mangelson *et al.* also use the BCH approximation during their derivations of the covariance of relative poses. The derivations present here then end up using this approximation twice. This could lead to increased inaccuracy between the derived covariance and actual covariance.

Secondly, to use the BCH formula in Equation (3.25), an assumption was made that the noise always is of much smaller magnitude than the rotation and that the approximation error as such is low. This, however, is not always the case for ship data. Since the measurements are available nearly continuously in time, then any relative pose computed between two absolute poses close in time will be nearly the identity, and its rotation vector is as such very small. However, rotations close in time also coincide with datapairs of low excitation, as will be shown in Section 4.3. As such, feeding these combinations of absolute rotations to the Hand-Eye calibration problem should be avoided altogether.

Thirdly, it is also of interest to investigate the effect of inserting the iteratively approximated value of $\mathbf{R}_\mathrm{X}$ into the expression of the covariance Equation (3.15). Using the estimate more correctly reflects the derived covariance of the residual, but one can imagine this leading to poor numeric properties when this estimate is updated at each step of the optimization.

Fourthly, note how the true rotations $\bar{\mathbf{R}}(t)$ appear in Equations (3.26) and (3.27), while we only have knowledge of their noisy counterparts. Using the noisy measurements in the covariance could also impact performance.

Lastly, in this work, the noise on measured the ship and camera rotations were assumed independent. If this methodology is to be used on a real-world dataset where the covariances have been estimated then this assumption must be challenged. One can for instance imagine both inertial measurement units and egomotion algorithms being affected similarly by the ship being hit by sudden waves. For the simulations in Section 4.2 only synthetic datasets are tested, and as such the noise on both ship and camera can be made independent by construction.

These considerations should impact performance but also point in the direction of how the performance of the covariance compensation methods should be tested. We will therefore in Section 4.2 test different variations of these assumptions.

3.3 Information and Convexity of Hand-Eye-data

The ease at which a Hand-Eye calibration problem can be solved is dependent on the input-data. As explained in Sections 2.6.3 and 3.1.3, both the observability and the covariance of the estimate are highly dependent on the parallelity of the rotation vectors of input relative poses. These explanations, as well as those present in literature, sadly do not open the door to answering numerical questions on the topic of data-selection for Hand-Eye calibration. Questions such as "how much is the uncertainty of the estimate expected to decrease if this specific datapoint is included in the dataset?", and "out of all the available absolute poses, which combination of relative poses will result in lowest uncertainty in the estimate?" remain unanswered.

This section introduces a metric for numerically quantifying the information of any single datapair $(\boldsymbol{\alpha}_i, \boldsymbol{\beta}_i)$ relative to an entire dataset $\left\{(\boldsymbol{\alpha}_i, \boldsymbol{\beta}_i)\right\}_{i \in \mathcal{T}}$, where $\mathcal{T}$ is used to denote the set of all measurement-timestamps. This is done by utilizing the Taylor expansion to build an approximation of the cost-function's Hessian about the ground-truth extrinsics. The resulting *Hand-Eye information* is used as a weighting scheme in the nonlinear least squares Hand-Eye solver, as well as data selection algorithms.

The derivation is based on the Park-Martin cost-function. Recall that the Park-Martin formulation of the Hand-Eye calibration problem is an isomorphism of the original formulation, meaning all properties of the former apply to the latter. This means the developed method is compatible with any other Hand-Eye solver.

3.3.1 The approximate Hessian of the Park-Martin cost-function

The first step in building the proposed metric is to approximate the Hessian of the Park-Martin cost-function. This is done following the method described in Section 2.4. Restating briefly: The Hessian of the cost-function can be approximated by the square of Jacobians of the measurement-prediction functions.

We define the block-vector function of measurements as

$$\mathbf{y} := \begin{bmatrix} \boldsymbol{\alpha}_1 \\ \vdots \\ \boldsymbol{\alpha}_N \end{bmatrix}, \tag{3.28}$$

and the block-vector function of measurement-predictions as

$$\mathbf{f}(\mathbf{x}) = \mathbf{f}(\mathbf{R}) := \begin{bmatrix} \mathbf{R}\boldsymbol{\beta}_1 \\ \vdots \\ \mathbf{R}\boldsymbol{\beta}_N \end{bmatrix}. \tag{3.29}$$

Taking the Taylor-expansion of $\mathbf{f}$ as a function of a rotation is possible [25], but cumbersome. Moreover, in this thesis, the optimization is done over the angle-axis form of $\mathbf{R}$, and as such it is reasonable to use this parameterization here as well. Then $\mathbf{x}_0 = \boldsymbol{\omega}_\mathrm{X} = \mathrm{Log}(\mathbf{R}_\mathrm{X})$ is chosen as the ground-truth sensor orientation, meaning $\mathrm{Exp}(\boldsymbol{\omega}_\mathrm{X})\boldsymbol{\beta}_i = \boldsymbol{\alpha}_i$ for all datapairs $(\boldsymbol{\alpha}_i, \boldsymbol{\beta}_i)$.

The Jacobian of a single prediction-function follows the chain-rule [25] and is therefore $\mathbf{J}^{\mathbf{R}\boldsymbol{\beta}_i}_{\boldsymbol{\omega}} = \mathbf{J}^{\mathbf{R}\boldsymbol{\beta}_i}_{\mathbf{R}}\mathbf{J}^{\mathbf{R}}_{\boldsymbol{\omega}}$. The first Jacobian is simply $\mathbf{J}^{\mathbf{R}\boldsymbol{\beta}_i}_{\mathbf{R}} = -\mathbf{R}\left[\boldsymbol{\beta}_i\right]_\times$, while the second term is the "right Jacobian", $\mathbf{J}_\mathrm{r}(\boldsymbol{\omega})$ of the SO(3) group. See Section 2.2.4 for more information on these Jacobians.

The Jacobian of the stacked measurement-prediction functions is then simply the column vector of stacked Jacobians. Inserting the ground truth-value $\boldsymbol{\omega} = \boldsymbol{\omega}_\mathrm{X} = \mathrm{Log}(\mathbf{R}_\mathrm{X})$ gives

$$\mathbf{J}_0 = \begin{bmatrix} -\mathbf{R}_\mathrm{X}\left[\boldsymbol{\beta}_1\right]_\times \mathbf{J}_\mathrm{r}(\boldsymbol{\omega}_\mathrm{X}) \\ \vdots \\ -\mathbf{R}_\mathrm{X}\left[\boldsymbol{\beta}_N\right]_\times \mathbf{J}_\mathrm{r}(\boldsymbol{\omega}_\mathrm{X}) \end{bmatrix}, \tag{3.30}$$

which in turn means that the approximate Hessian is on the form

$$\begin{aligned} \mathbf{H} &\approx \mathbf{J}_0^\top \mathbf{J}_0 \\ &= \begin{bmatrix} -(\mathbf{R}_\mathrm{X}\left[\boldsymbol{\beta}_1\right]_\times \mathbf{J}_\mathrm{r}(\boldsymbol{\omega}_\mathrm{X}))^\top & \dots & -(\mathbf{R}_\mathrm{X}\left[\boldsymbol{\beta}_N\right]_\times \mathbf{J}_\mathrm{r}(\boldsymbol{\omega}_\mathrm{X}))^\top \end{bmatrix} \begin{bmatrix} -\mathbf{R}_\mathrm{X}\left[\boldsymbol{\beta}_1\right]_\times \mathbf{J}_\mathrm{r}(\boldsymbol{\omega}_\mathrm{X}) \\ \vdots \\ -\mathbf{R}_\mathrm{X}\left[\boldsymbol{\beta}_N\right]_\times \mathbf{J}_\mathrm{r}(\boldsymbol{\omega}_\mathrm{X}) \end{bmatrix} \\ &= \mathbf{J}_\mathrm{r}(\boldsymbol{\omega}_\mathrm{X})^\top \left(\left[\boldsymbol{\beta}_1\right]_\times^\top \mathbf{R}_\mathrm{X}^\top \mathbf{R}_\mathrm{X}\left[\boldsymbol{\beta}_1\right]_\times + \dots + \left[\boldsymbol{\beta}_N\right]_\times^\top \mathbf{R}_\mathrm{X}^\top \mathbf{R}_\mathrm{X}\left[\boldsymbol{\beta}_N\right]_\times\right)\mathbf{J}_\mathrm{r}(\boldsymbol{\omega}_\mathrm{X}) \\ &= \mathbf{J}_\mathrm{r}(\boldsymbol{\omega}_\mathrm{X})^\top \left(\left[\boldsymbol{\beta}_1\right]_\times^\top \left[\boldsymbol{\beta}_1\right]_\times + \dots + \left[\boldsymbol{\beta}_N\right]_\times^\top \left[\boldsymbol{\beta}_N\right]_\times\right)\mathbf{J}_\mathrm{r}(\boldsymbol{\omega}_\mathrm{X}). \end{aligned} \tag{3.31}$$

Summarizing: The above calculations give that the approximate Hessian of the Park-Martin cost-function, with center in the ground-truth extrinsics, is on the form seen in Equation (3.32).

$$\mathbf{H} \approx \mathbf{J}_0^\top \mathbf{J}_0 = \mathbf{J}_r(\boldsymbol{\omega}_\mathrm{X})^\top \left(\sum_{i=1}^{N} \left[\boldsymbol{\beta}_i\right]_\times^\top \left[\boldsymbol{\beta}_i\right]_\times\right) \mathbf{J}_r(\boldsymbol{\omega}_\mathrm{X}) \tag{3.32}$$

In the derivations, the unknown ground-truth extrinsics $\mathbf{R}_\mathrm{X}$ cancelled nicely. The right Jacobian at the equally unknown ground-truth rotation vector $\boldsymbol{\omega}_\mathrm{X}$, however, did not. Unfortunately, it is not the case that the transpose of the right Jacobian $\mathbf{J}_r^\top(\boldsymbol{\omega}_\mathrm{X})$ is equal to its inverse $\mathbf{J}_r(\boldsymbol{\omega}_\mathrm{X})^{-1}$. This would have been very practical, as this would mean the effect of the right Jacobian was simply a change of basis. To circumvent the right Jacobian at the groud-truth being unknown, further analysis focuses on the quadratic form $\mathbf{x}^\top\mathbf{H}\mathbf{x}$. Comparing this with Equation (3.32) reveals Equation (3.33), where $\tilde{\mathbf{H}}$ is the sum of the squared skew-symmetric matrices.

$$\begin{aligned} \mathbf{x}^\top\mathbf{H}\mathbf{x} &= \mathbf{x}^\top\mathbf{J}_r(\boldsymbol{\omega}_\mathrm{X})^\top\tilde{\mathbf{H}}\mathbf{J}_r(\boldsymbol{\omega}_\mathrm{X})\mathbf{x} \\ &= (\mathbf{J}_r(\boldsymbol{\omega}_\mathrm{X})\mathbf{x})^\top\tilde{\mathbf{H}}(\mathbf{J}_r(\boldsymbol{\omega}_\mathrm{X})\mathbf{x}) \\ :&= \tilde{\mathbf{x}}^\top\tilde{\mathbf{H}}\tilde{\mathbf{x}} \end{aligned} \tag{3.33}$$

The right Jacobian has full rank, a proof of which is given in Appendix A.1. Therefore the effect of the right Jacobian is simply some transformation of any input $\mathbf{x}$ to the quadratic form $s(\mathbf{x}) = \mathbf{x}^\top \mathbf{H}\mathbf{x} = \tilde{\mathbf{x}}^\top \tilde{\mathbf{H}}\tilde{\mathbf{x}}$. That is, the right Jacobian will not map any vectors to zero, thanks to it being full rank. Still, it is not known whether different inputs will get different lengths and whether the right Jacobian can introduce skewness to the coming derivations.

For the time being, we hope that the effect of the right-Jacobian is not destructive for the results derived further, an assumption which is tested in Section 4.3. For all further analyses, the right Jacobian evaluated at ground-truth extrinsics is therefore omitted. And with this, the approximate Hessian about the ground-truth extrinsics are entirely described by known objects, that being the datapoints $\boldsymbol{\beta}_i$.

We additionally note the *analytic* Jacobian (not Hessian) of the full cost-function is

$$\begin{aligned} F(\mathbf{R}) = F(\mathrm{Exp}(\boldsymbol{\omega})) &= \frac{1}{2}(\mathbf{y}-\mathbf{f}(\mathbf{R}))^\top(\mathbf{y}-\mathbf{f}(\mathbf{R})) \\ \mathbf{J}^F_{\boldsymbol{\omega}}(\boldsymbol{\omega}) &= (\mathbf{y}-\mathbf{f}(\mathbf{R}))^\top \begin{bmatrix} -\mathrm{Exp}(\boldsymbol{\omega})\left[\boldsymbol{\beta}_1\right]_\times \mathbf{J}_\mathrm{r}(\boldsymbol{\omega}) \\ \vdots \\ -\mathrm{Exp}(\boldsymbol{\omega})\left[\boldsymbol{\beta}_N\right]_\times \mathbf{J}_\mathrm{r}(\boldsymbol{\omega}) \end{bmatrix}. \end{aligned} \tag{3.34}$$

We can confirm that the Jacobian is zero as expected when evaluated in the ground-truth extrinsics $\mathbf{y} = \mathbf{f}(\mathbf{R}_X)$, in the case with no noise on the data. Deriving an analytic Hessian was also considered for this work. This results in an expression with, among other things, the Jacobian of Equation (3.30) which was deemed too difficult to evaluate. The approximate Hessian is therefore used exclusively further, but it is of course only an approximation.

3.3.2 Positive (semi)-definiteness of the approximate Hessian

As explained shortly in Section 2.4, the positive (semi)-definiteness of an optimization function's Hessian describes whether the function is convex. If the Hessian centered at the global minimum is found to be positive semi-definite, then this implies the existence of some search directions for which the cost may stay constant. This will, at best, lead to suboptimal performance of the optimization software and at worst may imply the existence of multiple nearby minima. It is therefore of interest to analyze the positive definiteness of the cost-function's Hessian, and characterize these properties based on quantitative properties of the input data. For the following derivations we define $\left[\boldsymbol{\beta}_i\right]_\times := \mathbf{S}_i$.

Firstly, we note the fact the product of any matrix and its transpose is symmetric, from the relationship
$(\mathbf{S}_i^\top \mathbf{S}_i)^\top = \mathbf{S}_i^\top (\mathbf{S}_i^\top)^\top = \mathbf{S}_i^\top \mathbf{S}_i$. Further, any product of a matrix and its transpose is positive semi-definite as well, see Equation (3.35).

$$\mathbf{x}^\top(\mathbf{S}_i^\top \mathbf{S}_i)\mathbf{x} = (\mathbf{x}^\top \mathbf{S}_i^\top)(\mathbf{S}_i\mathbf{x}) = (\mathbf{S}_i\mathbf{x})^\top(\mathbf{S}_i\mathbf{x}) = ||\mathbf{S}_i\mathbf{x}||_2^2 \geq 0,\ \forall \mathbf{x} \in \mathbb{R}^3 \tag{3.35}$$

Then the approximate Hessian is symmetric and positive semi-definite as well. From the last equality, it is possible to recognize that

Proposition 3. *The product of a* 3×3 *skew-symmetric matrix and its transpose is positive definite with respect to all vectors* $\mathbf{x} \in \mathbf{R}^3 \setminus \mathrm{sp}(\{\boldsymbol{\beta}_i\})$*, where* sp *denotes the span of a set of vectors.*

Proof. From Equation (3.35) it is clear that $\mathbf{S}_i^\top \mathbf{S}_i$ is indefinite for some vector $\mathbf{x}$ if and only if $\mathbf{x}$ is in the null-space of $\mathbf{S}_i$, since $||\mathbf{S}_i\mathbf{x}|| = 0 \iff \mathbf{S}_i\mathbf{x} = \mathbf{0}$ from the definition of a norm. Since $\mathbf{S}_i\mathbf{x} = \left[\boldsymbol{\beta}_i\right]_\times \cdot \mathbf{x} = \boldsymbol{\beta}_i \times \mathbf{x} = ||\boldsymbol{\beta}_i|| \cdot ||\mathbf{x}|| \sin\left(\angle(\boldsymbol{\beta}_i, \mathbf{x})\right)\mathbf{n}$ with $\mathbf{n}$ being the vector normal to both $\boldsymbol{\beta}_i$ and $\mathbf{x}$, then $\mathbf{S}_i\mathbf{x} = \mathbf{0} \iff \mathbf{x} \parallel \boldsymbol{\beta}_i$, excluding the trivial cases of $\mathbf{x} = \mathbf{0}$ and $\boldsymbol{\beta}_i = \mathbf{0}$. □

This leads to the next result, which is pertinent for our case where multiple, or at least two, skew-symmetric matrices are summed.

Proposition 4. *The sum of two transposed-squared skew-symmetric matrices,* $\left[\boldsymbol{\beta}_i\right]_\times^\top \left[\boldsymbol{\beta}_i\right]_\times + \left[\boldsymbol{\beta}_j\right]_\times^\top \left[\boldsymbol{\beta}_j\right]_\times$*, is positive definite if and only if* $\boldsymbol{\beta}_i \notin \mathrm{sp}(\boldsymbol{\beta}_j)$

Proof. If $\boldsymbol{\beta}_i \notin \mathrm{sp}(\boldsymbol{\beta}_j)$ then the result follows directly from Proposition 3. For the back-implication, if $\left[\boldsymbol{\beta}_i\right]_\times^\top \left[\boldsymbol{\beta}_i\right]_\times + \left[\boldsymbol{\beta}_j\right]_\times^\top \left[\boldsymbol{\beta}_j\right]_\times$ is not positive definite, then it implies the existence of some $\mathbf{x} \in \mathbb{R}^3,\ \mathbf{x} \neq \mathbf{x}$ such that

$$\begin{aligned} \mathbf{x}^\top \left(\left[\boldsymbol{\beta}_i\right]_\times^\top \left[\boldsymbol{\beta}_i\right]_\times + \left[\boldsymbol{\beta}_j\right]_\times^\top \left[\boldsymbol{\beta}_j\right]_\times\right)\mathbf{x} = 0 \\ \mathbf{x}^\top \left[\boldsymbol{\beta}_i\right]_\times^\top \left[\boldsymbol{\beta}_i\right]_\times \mathbf{x} + \mathbf{x}^\top \left[\boldsymbol{\beta}_j\right]_\times^\top \left[\boldsymbol{\beta}_j\right]_\times \mathbf{x} = 0 \\ \implies \mathbf{x}^\top \left[\boldsymbol{\beta}_i\right]_\times^\top \left[\boldsymbol{\beta}_i\right]_\times \mathbf{x} = \mathbf{x}^\top \left[\boldsymbol{\beta}_j\right]_\times^\top \left[\boldsymbol{\beta}_j\right]_\times \mathbf{x} = 0, \end{aligned} \tag{3.36}$$

since each of the terms of the sum is positive semi-definite. But from Proposition 3 this requires both $\mathbf{x} \parallel \boldsymbol{\beta}_i$ and $\mathbf{x} \parallel \boldsymbol{\beta}_j$, and as such $\boldsymbol{\beta}_i \in \mathrm{sp}(\boldsymbol{\beta}_j)$. □

Comparing the results above to the approximate Hessian in Equation (3.32) we find that the cost function has zero curvature in the direction of $\tilde{\boldsymbol{\beta}}_i = \mathbf{J}_r(\boldsymbol{\omega}_\mathrm{X})\boldsymbol{\beta}_i$ around the ground-truth, when two parallel datapoints are used. Recalling the analytic expression for the Jacobian of the cost-function, Equation (3.34), one can see that the cost-function additionally has zero gradient along $\tilde{\boldsymbol{\beta}}_i$. This combined with the zero curvature means the cost is flat locally along this direction, forming a one-dimensional valley for which any estimate is equally as valid. This supports the findings in Section 3.1, where the $C = 0$ level sets - that being centered at the ground-truth - were found to be one-dimensional.

Let us refine these results by moving away from the general notion of "curvature" and positive-definitiveness, and attempt to assign some numerical values to the properties of the cost-function about the true extrinsics.

3.3.3 Eigenspace of Hessian and bounds on the quadratic form

The Hessian matrix enters the cost-function through the Taylor-expansion, where it appears in quadratic form $\mathbf{x}^\top \tilde{\mathbf{H}}\mathbf{x}$. In the last subsection, we explored under which conditions this product is strictly positive, but let us also attempt to give the numerical value of the product some meaning. Once more we ignore the right Jacobian and focus on the innermost product.

$$\begin{aligned}
\mathbf{x}^\top \tilde{\mathbf{H}}\mathbf{x} &= \mathbf{x}^\top \left(\sum_{i=1}^{N} \left[\boldsymbol{\beta}_i\right]_\times^\top \left[\boldsymbol{\beta}_i\right]_\times \right) \mathbf{x} \\
&= \mathbf{x}^\top \left(\sum_{i=1}^{N} \mathbf{S}_i^\top \mathbf{S}_i \right) \mathbf{x} \\
&= \sum_{i=1}^{N} \mathbf{x}^\top \mathbf{S}_i^\top \mathbf{S}_i \mathbf{x} \\
&= \sum_{i=1}^{N} ||\mathbf{S}_i \mathbf{x}||_2^2
\end{aligned} \tag{3.37}$$

This is the most refined answer we can get for any arbitrary vector $\mathbf{x}$ at this point, as the exact value will depend on the direction of the vector $\mathbf{x}$ and its length. However, the results in Section 3.3.2 can be used to establish a lower bound of 0 in the case when all datapoints are parallel. Using the property of the matrix norm of $||\mathbf{A}\mathbf{x}||_p \leq ||\mathbf{A}||_p \cdot ||\mathbf{x}||_p$, an upper limit of $\sum_{i=1}^{N} ||\mathbf{S}_i||_2^2 \cdot ||\mathbf{x}||_2^2$ can be established as well. Defining $\mathbf{x}$ to be some unit-sized step simplifies this expression to the following:

$$0 \leq \mathbf{x}^\top \tilde{\mathbf{H}}\mathbf{x} \leq \sum_{i=1}^{N} ||\mathbf{S}_i||_2^2 = \sum_{i=1}^{N} \lambda_{\max}(\mathbf{S}_i^\top \mathbf{S}_i), \tag{3.38}$$

where the last equality follows from the definition of the 2-norm over matrices. Quantifying the upper bound requires finding the eigenvalues of the matrix $\mathbf{S}_i^\top \mathbf{S}_i$ for any given $\boldsymbol{\beta}_i$. Since the matrix has been shown to be symmetric it is expected for its eigenspaces to be orthogonal, and since the matrix is positive semi-definite the eigenvalues must be non-negative.

From Proposition 3 we know $\boldsymbol{\beta}_i$ to span the null-space of $\mathbf{S}_i^\top \mathbf{S}_i$, which means $\lambda_{\min} = 0$ must be an eigenvalue of $\mathbf{S}_i^\top \mathbf{S}_i$ with eigenspace $\text{sp}(\{\boldsymbol{\beta}_i\})$. For the other two eigenvalues, a geometric argument can be made. Consider Figure 3.7. Performing the product $\mathbf{S}_i^\top \mathbf{S}_i \mathbf{v} = -\left[\boldsymbol{\beta}_i\right]_\times \left[\boldsymbol{\beta}_i\right]_\times \mathbf{v} = -\boldsymbol{\beta}_i \times (\boldsymbol{\beta}_i \times \mathbf{v})$ will result in a vector lying in the plane spanned by $\boldsymbol{\beta}_i$ and $\mathbf{v}$. This is because $\boldsymbol{\beta}_i \times \mathbf{v}$ must be perpendicular to the plane spanned by $\boldsymbol{\beta}_i$ and $\mathbf{v}$, while $\boldsymbol{\beta}_i \times (\boldsymbol{\beta}_i \times \mathbf{v})$ must be perpendicular to the plane spanned by $\boldsymbol{\beta}_i$ and $\boldsymbol{\beta}_i \times \mathbf{v}$. Alternatively, one can come to this conclusion by applying the right-hand rule to $\boldsymbol{\beta}_i$ and $\mathbf{v}$ twice. It is also clear that if $\mathbf{v} \perp \boldsymbol{\beta}_i$ then $(\mathbf{S}_i^\top \mathbf{S}_i \mathbf{v}) \parallel \mathbf{v}$, meaning $\mathbf{v}$ is an eigenvector. Its eigenvalue will be

$||\boldsymbol{\beta}_i \times (\boldsymbol{\beta}_i \times \mathbf{v})|| = ||\boldsymbol{\beta}_i|| \cdot ||\boldsymbol{\beta}_i|| \cdot ||\mathbf{v}|| \implies \lambda_{\text{max}} = ||\boldsymbol{\beta}_i||^2$. Since this reasoning applies for any vector $\mathbf{v}$ perpendicular to $\boldsymbol{\beta}_i$ then it must be the case that all vectors in the plane perpendicular to $\boldsymbol{\beta}_i$ will be eigenvectors with this same eigenvalue.

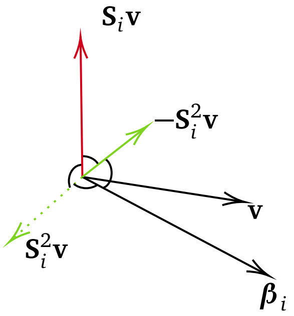


Figure 3.7: Illustration to help derivation of the eigenspaces of $\mathbf{S}_i^\top \mathbf{S}_i$. All angles marked in black are right angles.

Summarizing these results: $\mathbf{S}_i^\top \mathbf{S}_i$ has eigenspaces $\text{eig}_{\text{min}}(\mathbf{S}_i^\top \mathbf{S}_i) = \text{sp}(\{\boldsymbol{\beta}_i\})$ and $\text{eig}_{\text{max}}(\mathbf{S}_i^\top \mathbf{S}_i) = \text{sp}(\{\boldsymbol{\beta}_i\})^\perp$ with corresponding eigenvalues $\lambda_{\text{min}} = 0$ and $\lambda_{\text{max}} = ||\boldsymbol{\beta}_i||^2$. Thus any unit-length quadratic of the approximate Hessian is bounded by

$$0 \leq \mathbf{x}^\top \tilde{\mathbf{H}} \mathbf{x} \leq \sum_{i=1}^{N} ||\boldsymbol{\beta}_i||^2. \tag{3.39}$$

These derivations are useful as they enable a numerical description of the curvature of the cost-function, and therefore of the uncertainty associated with each search direction around the ground truth. The previous chapter concluded with the cost-function being convex for non-parallel data, and with the derivations in this chapter we are able to quantify this convexity and thus compare the contribution from different datapoints. Having such a quantification will further in this thesis enable the creation of "ranking" the datapoints in regards to their contribution in lowering the uncertainty of the estimates.

3.3.4 Continuity and monotony of the quadratic form

In Section 3.3.3 we derived bounds on the quadratic form $s(\mathbf{x}) = \mathbf{x}^\top \tilde{\mathbf{H}} \mathbf{x}$ which relates to the convexity, and therefore uncertainty, in different directions about the ground-truth. This was done by identifying the eigenspaces of the approximate Hessian, and finding the increase in cost due to convexity in the directions of said eigenspaces. If possible it is also of interest to inspect what happens in between the eigenspaces, thus obtaining knowledge of the cost-function's behavior for any input. The fact that the eigenspaces of each term $\mathbf{S}_i^\top \mathbf{S}_i$ are orthogonal makes this quite simple, as is shown in this section. It should be noted that the derivations in this section most likely have been derived elsewhere in literature, however, the author has not succeeded in finding any proof of this.

Let $\hat{\mathbf{u}}$, $\hat{\mathbf{v}}$ be an orthonormal basis of $\mathrm{sp}(\{\boldsymbol{\beta}_i\})^{\perp}$ and let $\hat{\boldsymbol{\beta}}_i$ be a unit-length basis of $\mathrm{sp}(\{\boldsymbol{\beta}_i\})$. Then $\hat{\mathbf{u}}$, $\hat{\mathbf{v}}$, $\hat{\boldsymbol{\beta}}_i$ is an unordered basis of $\mathbb{R}^3$. Any unit-length vector $\mathbf{x} \in S_1$ can be expressed as a linear combination of the three basis vectors as $\mathbf{x} = a\hat{\mathbf{u}} + b\hat{\mathbf{v}} + c\hat{\boldsymbol{\beta}}_i$, where $a^2 + b^2 + c^2 = 1$. Then

$$\begin{aligned}
\mathbf{x}^{\top}\mathbf{S}_i^{\top}\mathbf{S}_i\mathbf{x} &= \mathbf{x}^{\top}(a\lambda_{\max}\hat{\mathbf{u}} + b\lambda_{\max}\hat{\mathbf{v}} + c\lambda_{\min}\hat{\boldsymbol{\beta}}_i) \\
&= a^2\lambda_{\max} + b^2\lambda_{\max} + c^2\lambda_{\min} \\
&= (1-c^2)\lambda_{\max} + c^2\lambda_{\min} \\
&= \lambda_{\max} - c^2(\lambda_{\max} - \lambda_{\min}),
\end{aligned} \tag{3.40}$$

which for our specific values of eigenvalues gives $\mathbf{x}^{\top}\mathbf{S}_i^{\top}\mathbf{S}_i\mathbf{x} = (1-c^2)||\boldsymbol{\beta}_i||^2$. For any given unit-length $\mathbf{x}$ and $\hat{\boldsymbol{\beta}}_i$, the value of c can be found as the orthogonal projection $c = \mathbf{x}^{\top}\hat{\boldsymbol{\beta}}_i = \frac{\mathbf{x}^{\top}\boldsymbol{\beta}_i}{||\boldsymbol{\beta}_i||}$. From this, the quadratic form of the complete approximate Hessian is then expressable as

$$\mathbf{x}^{\top}\tilde{\mathbf{H}}\mathbf{x} = \sum_{i=1}^{N}\mathbf{x}^{\top}\mathbf{S}_i^{\top}\mathbf{S}_i\mathbf{x} = \sum_{i=1}^{N}||\boldsymbol{\beta}_i||^2 - \left(\mathbf{x}^{\top}\boldsymbol{\beta}_i\right)^2. \tag{3.41}$$

This result is interesting in itself, and could potentially be explored further as an alternate method of regularizing information content in a dataset. However, this is not performed in this thesis, and Equation (3.41) will not be explored further.

Interestingly, Equation (3.40) also enables analysis of the monotony of the quadratic form of a single datapoint. Let $\mathbf{x}(t)$ be the parameterization of any geodesic along the unit sphere, starting at $\mathbf{x}(t_0) = \hat{\boldsymbol{\beta}}_i$, and ending at time $t = t_1$ at any point in $\mathrm{sp}(\{\boldsymbol{\beta}_i\})^{\perp}$. Then $\mathbf{x}(t)$ is parameterized by having $c = c(t) = \cos(t)$ and parameter span $[t_0, t_1] = [0, \frac{\pi}{2}]$. Then

$$\begin{aligned}
\frac{d}{dt}\left(\mathbf{x}(t)^{\top}\mathbf{S}_i^{\top}\mathbf{S}_i\mathbf{x}(t)\right) &= \frac{d}{dt}\lambda_{\max} - c(t)^2(\lambda_{\max} - \lambda_{\min}) \\
&= \frac{d}{dt}\lambda_{\max} - \cos(t)^2(\lambda_{\max} - \lambda_{\min}) \\
&= \sin(2t)(\lambda_{\max} - \lambda_{\min}) \geq 0 \ \forall t \in \left[0, \frac{\pi}{2}\right]
\end{aligned} \tag{3.42}$$

This last result shows that the effect of the quadratic form of a single datapoint is increasing monotonically from the eigenspace of the smallest eigenvalue to the eigenspace of the largest eigenvalue. This means the curvature associated with any single datapoint only ever increases away from the minimum. For practical purposes, this result can be interpreted as such: Say a dataset with an abundance of planar datapoints is being used, and the opportunity arises to slightly shift one of the datapoints away from the others, perhaps by replacing it with a datapoint acquired during more exciting maneuvers of the vessel. Then this result shows us that *any* shifting at all, no matter how small, away from the other datasets is advantageous with respect to making the uncertainty less homogeneous, since the measure of information associated with the shifting datapoint is monotonically increasing up until the datapoint is orthogonal to the others.

3.3.5 Information-based weighted nonlinear least squares

One challenge with using the Hand-Eye calibration framework for calibration of ship-mounted cameras has been noted earlier: Namely that the problem is unobservable for planar data. In practice, however, the movement performed by ships is never *perfectly* planar, with some rolling and pitching of the ship due to wave motion. This motion is still very small, and this results in bad convergence of any optimization procedure, especially when the presence of noise produces local minima. As shown in earlier work by the author, seen in Appendix C, the planar nature of the data will lead to badly posed optimization functions. The Park-Martin cost functions when using ship-data is convex, but not equally so in all directions.

Not all datapoints are created equal in the eyes of Hand-Eye calibration. And in the setting of planar data one would be more than happy to accept some datapoint that is non-parallel to the others. But even in the event that such a datapoint was available, it would be overshadowed by the others. The sum the of cost of all the parallel datapoints will be much higher than the cost associated with the singular non-parallel datapoint, meaning the cost function will still be unevenly convex.

If, however, the datapoints are weighted so that the more information-rich datapoint could contribute more to the cost, then it is expected for the cost-function to be more regular in its convexity, thus lending to better numerical properties.

The knowledge obtained and analysis performed so far in Section 3.3 has given a numerical way of quantifying the effect of different datapoints on the cost-function. This has been done by studying the approximate Hessian of the cost-function, and especially the quadratic form it is related to. This scalar metric of information is in this section used to introduce a weighting scheme for the nonlinear estimation problem with Park-Martin cost. Residuals weighted by this scheme will be more strongly present in the cost function if their associated data is “important”, in the sense that it offers more to observability than the other data.

The weighting scheme is motivated by working through the following example of how a high amount of similar data will overshadow singular, highly excited, datapoints:

Let $\{\boldsymbol{\beta}_i\}_{i=1,2,\ldots,N-1}$ be a set of parallel or nearly-parallel datapoints, and let $B := \mathrm{sp}(\boldsymbol{\beta}_1) \approx \mathrm{sp}(\boldsymbol{\beta}_2) \approx \cdots \approx \mathrm{sp}(\boldsymbol{\beta}_{N-1})$ be the approximate span of these datapoints. Let also $\boldsymbol{\beta}_N \perp \boldsymbol{\beta}_1$ be a singular “highly excited” (non-parallel) datapoint, with associated span $\mathrm{sp}(\boldsymbol{\beta}_N) \subset B^{\perp}$.

Recall that the approximate Hessian is calculated as $\tilde{\mathbf{H}} = \sum_{i=1}^{N} \mathbf{S}_i^{\top} \mathbf{S}_i$ and that the role of the Hessian is to bring information of curvature into the Taylor approximation through the previously analysed quadratic form $\mathbf{x}^{\top}\tilde{\mathbf{H}}\mathbf{x}$. Remember also from Section 3.3.3 that the nullspace of $\mathbf{S}_i^{\top}\mathbf{S}_i$ is $\mathrm{sp}(\boldsymbol{\beta}_i)$ and that the remaining eigenspace of $\mathbf{S}_i^\top \mathbf{S}_i$ is $\mathrm{sp}(\boldsymbol{\beta}_i)^\perp$ with double eigenvalue $||\boldsymbol{\beta}_i||^2$.

Let now $\mathbf{x} \in \mathrm{sp}(\boldsymbol{\beta}_N) \subset B^\perp$, $||\mathbf{x}|| = 1$ be any unit-length vector in the span of $\boldsymbol{\beta}_N$, thereby being in the positive eigenspace of $\mathbf{S}_i^\top \mathbf{S}_i$ when $i \neq N$. Then

$$\begin{aligned} \mathbf{x}^\top \tilde{\mathbf{H}} \mathbf{x} &= \mathbf{x}^\top \left(\sum_{i=1}^{N} \mathbf{S}_i^\top \mathbf{S}_i \right) \mathbf{x} \\ &= \sum_{i=1}^{N} \left\| [\boldsymbol{\beta}_i]_\times \mathbf{x} \right\|^2 \\ &= \sum_{i=1}^{N} \left\| \boldsymbol{\beta}_i \right\|^2 \sin(\angle \boldsymbol{\beta}_i, \mathbf{x})^2 \\ &= \sum_{i=1}^{N-1} \left\| \boldsymbol{\beta}_i \right\|^2 + 0. \end{aligned} \tag{3.43}$$

Note that in the last line, the sum only goes up to $N-1$, since $\mathbf{x}$ is in the null-space of the last datapoint's skew-symmetric matrix $\left[\boldsymbol{\beta}_N\right]_\times^\top \left[\boldsymbol{\beta}_N\right]_\times$. With this we have a numeric value of the curvature of the cost-function along the eigenspace of the "common" type of datapoint, that being the planar data.

Let now $\tilde{\mathbf{x}} \in B \subset \mathrm{sp}(\boldsymbol{\beta}_N)^\perp$, $||\tilde{\mathbf{x}}|| = 1$ be a vector in the eigenspace of the "uncommon" datapoint $\boldsymbol{\beta}_N$. Then the same quadratic form will become

$$\begin{aligned} \tilde{\mathbf{x}}^\top \tilde{\mathbf{H}} \tilde{\mathbf{x}} &= \tilde{\mathbf{x}}^\top \left(\sum_{i=1}^{N} \mathbf{S}_i^\top \mathbf{S}_i \right) \tilde{\mathbf{x}} \\ &= \sum_{i=1}^{N} \left\| [\boldsymbol{\beta}_i]_\times \tilde{\mathbf{x}} \right\|^2 \\ &= \left\| \boldsymbol{\beta}_N \right\|^2 + \sum_{i=1}^{N-1} 0. \end{aligned} \tag{3.44}$$

If one makes the additional soft assumption that $\frac{1}{N-1} \sum_{i=1}^{N-1} \left\| \boldsymbol{\beta}_i \right\|^2 \approx \left\| \boldsymbol{\beta}_N \right\|^2$, meaning that the singular orthogonal datapoint is not exceptionally larger than any of the other datapoints, we see that

$$\mathbf{x}^\top \tilde{\mathbf{H}} \mathbf{x} \approx (N-1) \left\| \boldsymbol{\beta}_N \right\|^2 = (N-1) \cdot \tilde{\mathbf{x}}^\top \tilde{\mathbf{H}} \tilde{\mathbf{x}}. \tag{3.45}$$

The value of the quadratic form is $N-1$ times larger in the direction of the common datapoint than the uncommon one. For an iterative optimization scheme attempting to minimize the cost-function, this discrepancy leads to ill-conditioning [20], meaning slow and inaccurate convergence. This example then shows how even in the presence of a single datapoint which provides orthogonality from the others, and thereby observability in theory, the actual convexity is much smaller along the span of the uncommon datapoint and thereby making its contribution insignificant compared to the other directions. The example also shows how this problem is compounded when the amount of planar datapoints is large. A possible solution to this would then be to include fewer of the parallel datapoints, but estimation in the presence of noise is generally expected to be better when as many datapoints as possible are used.

To counteract this issue, it would be beneficial to weight the residuals with high-information data more strongly than the residuals with data that does not provide much new info. To this end and motivated by the preceding example, the weighting function in Equation (3.46) is proposed.

$$w_i = \boldsymbol{\beta}_i^\top \left(\sum_{j=1}^{N} [\boldsymbol{\beta}_j]_\times^\top [\boldsymbol{\beta}_j]_\times \right) \boldsymbol{\beta}_i = \boldsymbol{\beta}_i^\top \tilde{\mathbf{H}} \boldsymbol{\beta}_i \tag{3.46}$$

The rationale behind the weighting scheme is as follows. The eigenspace and null-space of any single $\mathbf{S}_i^\top \mathbf{S}_i$-term are orthogonal. This means that the scalar value $w_i = \boldsymbol{\beta}_i^\top \left(\sum_{j=1}^{N} \mathbf{S}_j^\top \mathbf{S}_j \right) \boldsymbol{\beta}_i$ is maximized when $\boldsymbol{\beta}_i$ is as orthogonal to as many of the datapoints $\left\{ \boldsymbol{\beta}_j \right\}_{j \neq i}$ as possible. Firstly, this ensures that uncommon, important datapoints are assigned a high weighting factor, by virtue of being different from as many of the other datapoints as possible. Secondly, the weight assigned to two perfectly parallel datapoints is zero, correctly reflecting how perfectly parallel data does not offer anything in terms of observability and therefore just as easily could be ignored. The weighting scheme punishes datapoints whose span already is present in the optimization problem.

The weighting scheme is also inspired by the previously presented example. We will now see how using the scheme affects the quadratic form in the previously inspected directions of $\tilde{\mathbf{x}} \in B = \mathrm{sp}(\boldsymbol{\beta}_1) \approx \cdots \approx \mathrm{sp}(\boldsymbol{\beta}_{N-1})$ and $\mathbf{x} \in \mathrm{sp}(\boldsymbol{\beta}_N)$. We begin the example again by calculating the weights, and realizing that the weights for the first $N-1$ datapoints are exactly equal to

$$\begin{aligned}
w_i &= \boldsymbol{\beta}_i^\top \left(\sum_{j=1}^{N} [\boldsymbol{\beta}_j]_\times^\top [\boldsymbol{\beta}_j]_\times \right) \boldsymbol{\beta}_i \\
w_i &= (||\boldsymbol{\beta}_i|| \cdot \tilde{\mathbf{x}})^\top \left(\sum_{j=1}^{N} [\boldsymbol{\beta}_j]_\times^\top [\boldsymbol{\beta}_j]_\times \right) (\tilde{\mathbf{x}} \cdot ||\boldsymbol{\beta}_i||) \\
w_i &= \left\|\boldsymbol{\beta}_i\right\|^2 \tilde{\mathbf{x}}^\top \tilde{\mathbf{H}} \tilde{\mathbf{x}} \\
w_i &= \left\|\boldsymbol{\beta}_i\right\|^2 \left\|\boldsymbol{\beta}_N\right\|^2, \quad \forall i \in 1 \ldots N-1,
\end{aligned} \tag{3.47}$$

using the fact that we already calculated $\tilde{\mathbf{x}}^\top \tilde{\mathbf{H}} \tilde{\mathbf{x}}$ previously in this example. Similarly we calculate the weight of the N-th datapoint to be

$$\begin{aligned}
w_N &= \boldsymbol{\beta}_N^\top \left(\sum_{j=1}^{N} \left[\boldsymbol{\beta}_j\right]_\times^\top \left[\boldsymbol{\beta}_j\right]_\times \right) \boldsymbol{\beta}_N \\
w_N &= (||\boldsymbol{\beta}_N|| \cdot \mathbf{x})^\top \left(\sum_{j=1}^{N} \left[\boldsymbol{\beta}_j\right]_\times^\top \left[\boldsymbol{\beta}_j\right]_\times \right) (\mathbf{x} \cdot ||\boldsymbol{\beta}_N||) \\
w_N &= \left\|\boldsymbol{\beta}_N\right\|^2 \mathbf{x}^\top \tilde{\mathbf{H}} \mathbf{x} \\
w_N &= \left\|\boldsymbol{\beta}_N\right\|^2 \sum_{i=1}^{N-1} \left\|\boldsymbol{\beta}_i\right\|^2 .
\end{aligned} \tag{3.48}$$

The Hessian of the cost function of weighted residuals $\mathbf{r}_i' = w_i \mathbf{r}_i$ is $\mathbf{H}' = \sum_{i=1}^{N} w_i \left[\boldsymbol{\beta}_i\right]_\times^\top \left[\boldsymbol{\beta}_i\right]_\times$. Taking the calculated weights and weighing the residuals, we see that the quadratic forms previously analyzed now become

$$\mathbf{x}^\top \mathbf{H}' \mathbf{x} = \sum_{i=1}^{N-1} \left\|\boldsymbol{\beta}_i\right\|^2 \left\|\boldsymbol{\beta}_i\right\|^2 \left\|\boldsymbol{\beta}_N\right\|^2 \approx (N-1) \left\|\boldsymbol{\beta}_N\right\|^6 \tag{3.49}$$

and

$$\tilde{\mathbf{x}}^\top \mathbf{H}' \tilde{\mathbf{x}} = \left\|\boldsymbol{\beta}_N\right\|^2 \left\|\boldsymbol{\beta}_N\right\|^2 \sum_{i=1}^{N-1} \left\|\boldsymbol{\beta}_i\right\|^2 \approx (N-1) \left\|\boldsymbol{\beta}_N\right\|^6 . \tag{3.50}$$

The calculations above are quite intricate, but the result they try to illustrate is as follows: The proposed weighting scheme makes the quadratic form, which prior to weighting was heavily skewed in favor of the common parallel data, now equal for both directions of data. The quadratic form has been regularized. This should, in theory, allow the optimization to more efficiently make use of the rich information presented by the singular highly excited datapoint.

It should be noted that the proposed weighting is entirely novel and the resultant weighted estimator has no guarantee of being unbiased or optimal in any way. One could easily make any number of design choices differently, and still achieve the posted regularization of the quadratic form.

One such design choice could be to normalize the $\boldsymbol{\beta}_i$ before multiplying them with $\tilde{\mathbf{H}}$, thus calculating the weights as $w_i' = \left(\frac{\boldsymbol{\beta}_i}{||\boldsymbol{\beta}_i||}\right)^\top \tilde{\mathbf{H}} \left(\frac{\boldsymbol{\beta}_i}{||\boldsymbol{\beta}_i||}\right)$. This was not chosen in this thesis, as testing observed this to lead to inaccuracies. It was identified that this is because the SO(3)-logarithm is sensitive to input rotation matrices close to the identity. These kinds of rotation matrices occur when relative poses are calculated from two absolute pose measurements close in time. Inputting these rotations into the logarithm leads to very small vectors, pointing in wildly different directions and thus seeming more non-planar than their respective rotations actually are. When these vectors $\boldsymbol{\beta}_i$ are not normalized, the negative effect of their direction being inaccurate is counterweighted by their small size, so that the weight w_i overall is a small scalar. If the vectors are normalized, however, then vectors in wrongful directions will seem to be highly excited and therefore results in a large weighting of the residual.

An alternative to normalizing by the size of $\boldsymbol{\beta}_i$ could therefore be to instead normalize the weights of the non-normalized rotation vectors, based on the maximum weight in the dataset. This would be performed as $w_i' = w_i / \max_i w_i$. Additionally, as shown in Equations (3.49) and (3.50), the weight will also increase with dataset size. Thus when comparing the excitation present in two datasets of different sizes it will be more representative to compensate both datasets' information weights by their dataset size (minus one).

The proposed weighting scheme and some possible design choices are tested in Section 4.3.

3.4 Hand Eye data-selection

In the classical Hand-Eye calibration setup, the hand and eye are mounted to a robotic arm, meaning precise predetermined movements can be captured as data baseline for the estimation algorithms. As mentioned in Section 2.6.3, authors such as Tsai *et al.* [6] then give suggestions for how to choose predetermined poses to perform the estimation with the best numerical properties.

When performing Hand-Eye calibration as an auxiliary process on data gathered from a rig with some other main purpose than calibration, the predetermination of poses is most likely not possible. That is, a ship with mounted cameras generates poses through navigation and it will not be practical to “pause” the navigation of the ship to generate poses for calibration before continuing navigation. Further, not all possible poses are actually feasible for the system to undergo, and some subset of the feasible poses may be highly unlikely to actually occur.

To complicate matters further, when poses are obtained from modern navigational sensors, the measurements will be available with high frequency. This means a large amount of data is available at any given time and this amount only grows with time. Thus, any algorithm wanting to estimate based on the data must also handle the large number of datapoints somehow. For example, by only performing estimation on a subset of data, weighting the datapoints based on some measure of goodness or employing a receding horizon where old enough datapoints are ignored over time.

It could be beneficial to implement HE calibration in a real-time system for online updates of the extrinsic parameters, or even as a way to detect changes in camera orientation. In principle, the real-time application of Hand-Eye calibration of real-world data will require the algorithm to be able to produce meaningful estimates with whatever data is given. Remembering that the HE framework requires input to be relative poses based on two absolute poses, a natural question arises: Given a (possibly very large) dataset of non-predetermined absolute poses, how would one go about optimally choosing the pairs which make up a single relative pose fed to the algorithm?

For the reasons presented, it is therefore of interest to determine a strategy for choosing which pairs of absolute poses are to be combined into a single relative pose that the Hand-Eye calibration should be performed based on. We have previously in Section 3.3 explored the numerical properties of data when it comes to excitation, and it is possible to imagine a data-selection strategy also making use of this metric. The data-selection strategies proposed in Section 3.4 are in large based on qualitative assertions on what is good data for Hand-Eye calibration, and less so on a theoretical derivation of optimality.

Data selection or alternatively *methods for performing data selection* will in this thesis concern two things. Firstly is the issue of *data pairing*, meaning the pairing of absolute poses into relative poses. Secondly is the issue of *subset selection*, meaning the method used to reduce the set of all possible datapoints into a much smaller subset of datapoints actually fed to the Hand-Eye solver. A data selection method may perform both of these tasks simultaneously, or separately. If $2N$ datapoints are available, a total of $\mathrm{C}(2N, 2) = \frac{(2N)!}{(2N-2)!2!} = N(2N-1) \propto N^2$ unique pairs may be chosen when permutations of the poses are considered equal. Any algorithm for data-pairing that picks datapairs based on the entire set of possible combinations then must at least have a runtime of $\Omega(N^2)$ from simply constructing the set off all possible pairings. This may seem like bad news for the runtime of such an algorithm, until you consider that for subset selection the amount of possible partitions of $2N$ elements into subsets of size M grows astoundingly fast in N.[1] Therefore it is a natural choice to perform the pairing iteratively and greedily without taking into consideration the final subset of data this results in, as opposed to a holistic approach where both optimal selection of the final subset and the pairing of absolute poses within that subset are performed simultaneously.

3.4.1 Proposed data-selection strategies

In the following section, a number of data-selection strategies are proposed, and in Section 4.4 simulation results comparing these strategies are presented.

In the following subsections, $\mathcal{T}$ denotes the set of all measurement-timestamps $t_0, t_1, t_2, \dots t_N \in \mathcal{T}$. A data-selection strategy defines the pairs of absolute poses to make up a single pose. Then, since both relative ship-poses and relative camera-poses are fed pairwise into the Hand-Eye calibration problem, the data-selection strategy must be applied equally to both streams of data. For this reason, the strategies are here denoted by the way they pair *timestamps*, which are assumed common for both ship- and camera-poses.

All data relative first

The most simple way to combine absolute poses into a set of relative poses is to arbitrarily choose the first datapoint as the reference pose, and to compute all poses relative to this.

$$\mathcal{D}_{\text{rel.first}} = \left\{\mathbf{H}_{\text{nb}}(t_0)^{-1}\mathbf{H}_{\text{nb}}(t_i)\right\}_{t_i \in \mathcal{T} \setminus \{t_0\}} \tag{3.51}$$

If reality was such that only subset-selection – but not the pairing of absolute poses into relative poses – had any effect on estimation error, then the *all data relative first* data-selection strategy would be an optimal data-selection strategy. This is because in this hypothetical scenario, the choice to have all poses be calculated relative to the first datapoint would be truly arbitrary. However, as shown by example in Section 2.6.3, this scenario is strictly hypothetical, and data pairing will affect estimation error. The *all data relative first* strategy reflects the expected performance of any data-selection strategy where data pairing is not considered, and any strategy which considers data pairing is then expected to outperform *all data relative first*.

[1] See *Stirling numbers of the second kind.*

All possible pairs

Where the *all data relative first* strategy represents performing no data pairing at all, the *all possible pairs* data-selection strategy represents performing all possible data-pairings. The relative poses are generated using every unordered pair of timestamps (t_i, t_j), where t_i is defined without loss of generality to be a strictly earlier point in time than t_j. The resultant dataset is then also the superset of any possible subset selection strategy.

$$\mathcal{D}_{\text{all pairs}} = \left\{\mathbf{H}_{\text{nb}}(t_i)^{-1}\mathbf{H}_{\text{nb}}(t_j)\right\}_{t_i<t_j\in\mathcal{T}} \tag{3.52}$$

As mentioned earlier, least-squares methods over noisy data are generally expected to perform better the more data is available, by the noise being "averaged out". Intuition may then suggests that the *all possible pairs* strategy should perform the best of all strategies when residuals are weighted appropriately based on their level of noise and information.

The *all possible pairs* data-selection strategy will also be used as a baseline for other strategies presented in this section. As pointed out in [7], it is handy to preprocess the data by constructing all possible relative poses, as this allows for simply picking the best relative pose from the set of all possible instead of having to somehow evaluate the goodness of the absolute poses and then construct the best relative pose given two best absolute poses.

Tsai-Lenz score maxing

In Section 2.6.3, the work of Tsai *et al.* [6] was presented shortly. In their paper, they describe how the uncertainty of Hand-Eye estimate is dependent on, among other things, the angle between and magnitude of rotation vectors of relative poses. These criteria can then be used to construct a metric for evaluating any single datapairs' suitability for Hand-Eye calibration, and datapairs in the set of all possible datapairs can be chosen greedily based on this metric.

The proposed metric is as follows: Given some set of already chosen relative poses and their rotation vectors, $\mathcal{B} = \left\{\boldsymbol{\beta}_j\right\}_{j\in\mathcal{T}_{\text{chosen}}}$ and some datapoint $\boldsymbol{\beta}$ we wish to compare against the chosen data, the proposed *Tsai-Lenz score* is defined as

$$s(\boldsymbol{\beta};\ \mathcal{B}) = \frac{\|\boldsymbol{\beta}\|}{\pi} \cdot \frac{1}{|\mathcal{B}|} \sum_{\boldsymbol{\beta}_j\in\mathcal{B}} \left|\sin(\angle(\boldsymbol{\beta}, \boldsymbol{\beta}_j)\right|. \tag{3.53}$$

The score is the product of the query-datapoint's length, normalized to lie in the span $[0, 1]$, and the average sine of the angle between the query-data and already chosen data. This score is then a direct reflection of the relationship in Equation (2.53), Section 2.6.3. The score is maximized and Equation (2.53) is minimized when the query-datapoint has as large a length as possible and is as orthogonal to the chosen data as possible. The sine of the angle between rotation axes is evaluated in absolute value since, as explained in Section 2.6, the angle between rotation vectors associated with any relative pose is invariant under inversion. Explained mathematically $\mathrm{Log}(\mathbf{R}_q^\top \mathbf{R}_p) = -\mathrm{Log}(\mathbf{R}_p^\top \mathbf{R}_q)$, and observability only depends on non-parallelity, not the positiveness of angle between rotation vectors.

With this scoring defined, the *Tsai-Lenz score maxing* data selection strategy can be defined. The strategy is defined by starting with an empty set of chosen datapoints, and then greedily adding onto it with the datapoints which maximize the Tsai-Lenz score given the current set of chosen data. Pseudocode of the strategy is given in Algorithm 1. Here, $\mathcal{P}$ is the set of all possible rotation vectors, and $\mathcal{B}$ is the set of chosen data. Please note that the choice to use the rotations of the camera-poses to calculate the rotation vectors is as explained in Section 2.6.3 completely arbitrary.

Note that when implementing Algorithm 1, the procedure of noting which Homogeneous Transformation matrices **A** and **B** that are associated with the chosen set $\mathcal{B}$ has been omitted for posterity. This step, however, is of course important to implement, since the set of camera rotation vectors $\boldsymbol{\beta}_i$ alone is not sufficient data to feed into the Hand-Eye calibration problem.

Algorithm 1 Tsai-Lenz score-maxing data-selection

```
Require: ${(\mathbf{H}_{\mathrm{nb}}(t_i), \mathbf{H}_{\mathrm{mi}}(t_i))}_{t_i \in \mathcal{T}}$
  $\mathcal{P} \leftarrow \left\{\mathrm{Log}(\mathbf{R}_{\mathrm{mi}}(t_i)^\top \mathbf{R}_{\mathrm{mi}}(t_j))\right\}_{t_i < t_j}$
  $\mathcal{B} \leftarrow \emptyset$
  while $|\mathcal{B}| < M$ do
    $\boldsymbol{\beta}_i \leftarrow \max_{\boldsymbol{\beta} \in \mathcal{P}} s(\boldsymbol{\beta};\ \mathcal{B})$
    $\mathcal{B} \leftarrow \mathcal{B} \cup \left\{\boldsymbol{\beta}_i\right\}$
    Remove $\boldsymbol{\beta}_i$ from $\mathcal{P}$
  end while
```

Information maxing

In Section 3.3, a method is presented for quantifying the information any single datapair contributes to the overall estimate of the Hand-Eye calibration. With this metric in hand, a similar data-selection strategy to the *Tsai-Lenz score maxing* can be defined, only with greedy maximization of the HE information metric instead of the Tsai-Lenz score.

The method similarly begins by constructing the rotation vector of all possible relative poses, as well as initializing the set of chosen datapoints to consist only of the datapoint whose rotation vector has the longest length, $\mathcal{B} = \left\{\max_{\boldsymbol{\beta}} ||\boldsymbol{\beta}||\right\}$. Then, iteratively, the approximate Hessian presented in Section 3.3 is calculated for all datapoints chosen so far and the next datapoint is chosen greedily as the one maximizing the information metric. Pseudocode of the proposed data-selection strategy is given in Algorithm 2. Once more, the actual datapairs fed to the Hand-

Eye solver are found as the relative poses necessary to construct the chosen rotation vectors.

Algorithm 2 Hand-Eye information metric maximization data-selection

```
Require: $\{(\mathbf{H}_{\text{nb}}(t_i), \mathbf{H}_{\text{mi}}(t_i))\}_{t_i \in \mathcal{T}}$
  $\mathcal{P} \leftarrow \left\{\text{Log}(\mathbf{R}_{\text{mi}}(t_i)^\top \mathbf{R}_{\text{mi}}(t_j))\right\}_{t_i < t_j}$
  $\boldsymbol{\beta}_0 \leftarrow \max_{\boldsymbol{\beta} \in \mathcal{P}} ||\boldsymbol{\beta}||$
  $\mathcal{B} \leftarrow \left\{\boldsymbol{\beta}_0\right\}$
  $\tilde{\mathbf{H}} \leftarrow \left[\boldsymbol{\beta}_0\right]_\times^\top \left[\boldsymbol{\beta}_0\right]_\times$
  Remove $\boldsymbol{\beta}_0$ from $\mathcal{P}$
  while $|\mathcal{B}| < M$ do
    $\boldsymbol{\beta}_i \leftarrow \max_{\boldsymbol{\beta} \in \mathcal{P}} \boldsymbol{\beta}^\top \tilde{\mathbf{H}} \boldsymbol{\beta}$
    $\mathcal{B} \leftarrow \mathcal{B} \cup \left\{\boldsymbol{\beta}_i\right\}$
    $\tilde{\mathbf{H}} \leftarrow \tilde{\mathbf{H}} + \left[\boldsymbol{\beta}_i\right]_\times^\top \left[\boldsymbol{\beta}_i\right]_\times$
    Remove $\boldsymbol{\beta}_i$ from $\mathcal{P}$
  end while
```

Random pairs

Lastly, since the presented strategies attempt to choose data in a seemingly “intelligent” manner, then an absolute minimum baseline they must perform better than is to simply pair absolute poses at random. This strategy simply picks a random datapoint, then a second and checks that this pair has not already been drawn. If not then this pair is chosen and the algorithm continues until the dataset is at the desired size. If the pair has been chosen already, then the method draws a new pair.

Chapter 4

Simulation results

4.1 Simulation setup

4.1.1 The software

Previous works by the author, with preprint given in Appendix C, laid out a software pipeline for performing Hand-Eye calibration using ship-data. The pipeline reads the measured ship-poses from GPS and inertial navigation systems and constructs a local NED coordinate frame based on these. Simultaneously, captured images are fed to a Structure from Motion algorithm to estimate the camera poses. Finally, the ship and camera-poses are used to construct relative poses based on some data selection strategy and these relative poses are fed to a Hand-Eye solver of choice. The software is implemented in Python, using the open source SciPy [29] library for its general purpose nonlinear least squares solver. Two SfM libraries were tested in the previous works, OpenSfM [9] and COLMAP [30], with the latter being preferred due to higher accuracy and ease-of-use.

The same implementation of this software-pipeline was also used for simulations in this thesis.

4.1.2 The datasets

For performing simulations to investigate and substantiate the derived theory in this thesis, four datasets were used. These consisted of two computer-generated and four real-world datasets. A short presentation of each is given below.

Synthetic uniform

The *synthetic uniform* dataset is used as a baseline comparison to the other datasets. The poses are generated by drawing random poses uniformly over $\mathrm{SO}(3) \times [-L, L]^3$, for some span over possible positions L. Uniformly generating vectors over a closed interval of 3-dimensional space is a simple task, but drawing random rotation matrices with uniform probability is less straightforward. Drawing an axis and an angle uniformly does not uniformly cover the space of all rotations, and neither does drawing Euler-angles uniformly [31]. For this work, the method of Shoemake [32] is used, which is shown to result in a uniform distribution over SO(3). The ship-poses are drawn uniformly using the presented methods, and the camera-poses are constructed as the composition of generated ship-poses and an arbitrarily chosen set of constant extrinsic parameters. The generated poses are obviously not feasible for an actual ship with cameras, requiring for example the ship to be upside-down or pointed vertically, but the synthetic uniform dataset still serves as a useful baseline dataset. The synthetic uniform dataset represents movement with no underlying structure in said movement and no continuity between poses adjacent in time, being completely random from one timestamps to the next. This dataset is then also highly excited in the sense that the rotation axes of generated motions are decidedly non-parallel, see Section 2.6.3 for the importance of this.

Synthetic planar

The synthetic planar dataset is generated with the intention of closely resembling the actual movement of ships. The ship-poses undergo a random walk on yaw, the body Z-axis, with constant forward velocity and no up nor down velocity. Random perturbations in roll and pitch are added as noise to simulate waves, and the amplitude of these perturbations is controllable through changing the variance of the added noise. This results in ship-poses that operate mostly in the plane, but with small deviations away from purely planar motion. The camera-poses are generated identically to that of the synthetic uniform dataset; an arbitrary set of extrinsic parameters are chosen and multiplied with the ship-poses to generate the camera poses.

Additionally, as explained in Section 2.6.2, using the presented pipeline on real ship-data will cause the camera poses to be given in some unknown frame as well as with unknown scale on the translations. These aspects are also implemented in the synthetic planar dataset. The former is achieved by choosing an arbitrary HT $\mathbf{H}_{\mathrm{mn}}$ relating a simulated unknown mediary coordinate system generated by the SfM algorithm to NED. Premultiplying all generated camera-poses with this transform yields the camera-poses in this unknown frame. Then simply scaling the translations with the inverse of the desired scale parameter gives camera-poses structurally equivalent to those expected from real-world data. See also Equation (4.1) for a mathematical explanation.

$$\begin{gathered}\mathbf{H}_{\mathrm{ni}}(t_j) = \mathbf{H}_{\mathrm{nm}}\mathbf{f}_{\lambda}(\mathbf{H}_{\mathrm{mi}}(t_j)) \\ \Longleftrightarrow \\ \mathbf{H}_{\mathrm{mi}}(t_j) = \mathbf{f}_{1/\lambda}(\mathbf{H}_{\mathrm{nm}}^{-1}\mathbf{H}_{\mathrm{ni}}(t_j))\end{gathered} \tag{4.1}$$

One last feature of the synthetic planar dataset is the ability to model some sudden excitation of the system. It is imaginable that the ship may suddenly be hit by some larger-than-normal wave, and it is interesting to see how the different Hand-Eye solvers take advantage of the additional excitation.

Optional measurement noise is added to both synthetic datasets in the same way as explained in Section 2.2.2, and the covariance of each datapoint is then available for all timestamps. The two synthetic datasets are shown illustrated in Figure 4.1, with the figure taken from the project thesis this master's thesis is based on.

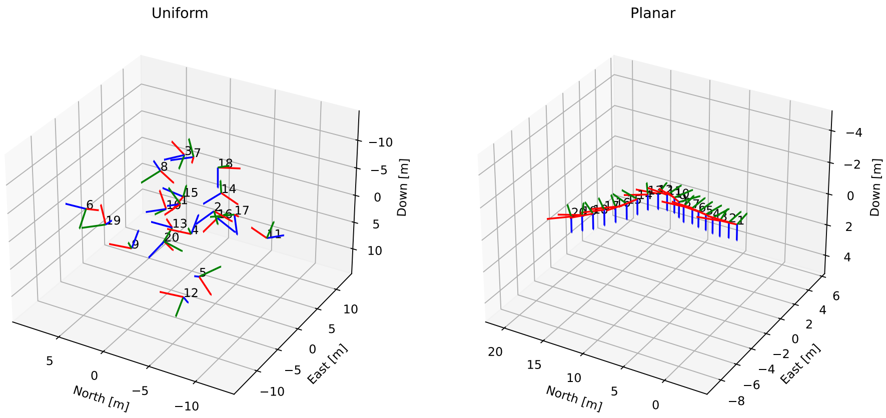


Figure 4.1: The 20 first generated poses in the synthetic uniform (left) and synthetic planar (right) datasets

Real-world

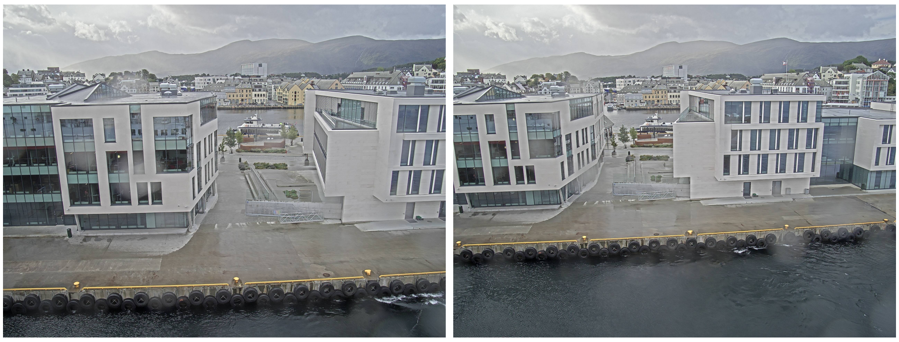
In addition to the two aforementioned synthetic datasets, two real-world datasets were also available for use in simulations in this thesis. These were supplied by Kongsberg Maritime (KM) as part of the research center SFI Autoship, a collaboration between Norwegian University of Science and Technology and various commercial partners, including both Kongsberg Maritime and SINTEF. The two datasets supplied by KM are dubbed the *KM weakly excited* and *KM strongly excited* datasets.

The KM weakly excited dataset was collected from a large passenger cruise ship fitted with camera rigs. The subset of the full dataset used in this thesis spanned 60 seconds as the cruise ship was leaving port. The closeness to land and buildings means a high amount of buildings and details could be tracked by the SfM algorithm, but this also meant waves that cause non-planar movement was minimal. This is compounded by the large size of the vessel, and the fact that cruise ships by design are made to rock as little as possible. The resulting motion is therefore almost perfectly planar, and the dataset is dubbed *weakly excited* therefrom.

The KM strongly excited dataset, on the other hand, was collected from a small research vessel performing rapid maneuvers outside the coast of Trondheim, Norway. The small size and rapid movements mean the captured movement is affected by waves to a large degree, thus motivating the name. The dataset was however captured far from land, meaning that there were fewer detectable features present in the images. Thus the SfM reconstruction of camera motions was shown in the project thesis leading up to this thesis to be more noisy than the KM weakly excited dataset.

If the results in this thesis are to be recreated for other real-world datasets, some considerations must be made. Firstly, the measurement systems fitted to the ship were done so for research purposes, and as such the measurements of ship-pose were much more accurate than should be expected from other similar systems. Secondly, a numerical uncertainty of the ground truth extrinsics was not known. This can make certain conclusions regarding results generated using this dataset hard to support, as it is not known whether the estimated orientation is wrong or if the ground-truth is wrong.

Figures 4.2 and 4.3 show illustrative images from both real-world datasets, for two chosen timestamps. Note how the horizon-line in Figure 4.3 changes much more drastically than in Figure 4.2. This behavior is consistent throughout both datasets, and therefore gives rise to their description as being *weakly* and *strongly* excited. Figures 4.2 and 4.3 are taken from the project thesis this thesis is based on.



(a) Taken at 07:30:00

(b) Taken at 07:30:17

Figure 4.2: Example images from the KM weakly excited dataset

4.1.3 The metric for evaluating estimation error

In the following section whenever an estimated camera-orientation is compared against the ground-truth extrinsics, the SO(3)-metric presented in Section 2.2.3 will be used. The resulting angle between the compared rotations is scaled to degrees for easier interpretation.

(a) Taken at 08:14:20 **(b)** Taken at 08:14:37

Figure 4.3: Example images from the KM strongly excited dataset

4.2 Covariance compensation

To accurately reflect the real-world scenario when performing covariance compensation, the setup described in Section 3.2 was implemented in software. Noise with locally defined covariances was added to the synthetic planar dataset and propagation of these covariances through the chosen relative poses was performed. Performing tests with synthetic noise was done to get an as accurate test of the proposed covariance compensation as possible, as well as due to the fact that covariance estimates for the real-world datasets were not available. Also, the used SfM pipelines did not produce uncertainties of the constructed camera motions either. Previous work by the author found the standard Park-Martin cost-function able to estimate extrinsics within 2° of the ground-truth when performed over the real-world dataset with unknown noise level. Therefore, noise was added to the synthetic planar dataset incrementally until the optimization gave a mean error of about 2°. This level of noise then became the baseline noise level, and levels slightly above and below this level were tested.

Correlation between measurements of the same frame's rotation at different timestamps, that is between ship rotations at timestamp t_j and t_k or between camera rotations at t_j and t_k, were not implemented. This was initially done for simplicity, with the intent of adding it later, but as the results in this section turned out much differently than expected, the energy was instead spent on debugging the behavior of the methods.

First, a simple comparison of performing no covariance compensation and performing compensation with the full group-theoretic covariance based on relative poses' covariance presented in Section 3.2 was performed. This was done by performing 80 runs of optimizing the standard Park-Martin residual as well as the covariance-compensated Park-Martin residual, each with a different instance of the same synthetic planar dataset with random covariances and noise realization for each run. Each run also used a random initial condition to the optimization, within 0.3 rad $\approx 17^\circ$ standard deviation of the ground-truth rotation vector. Ad-

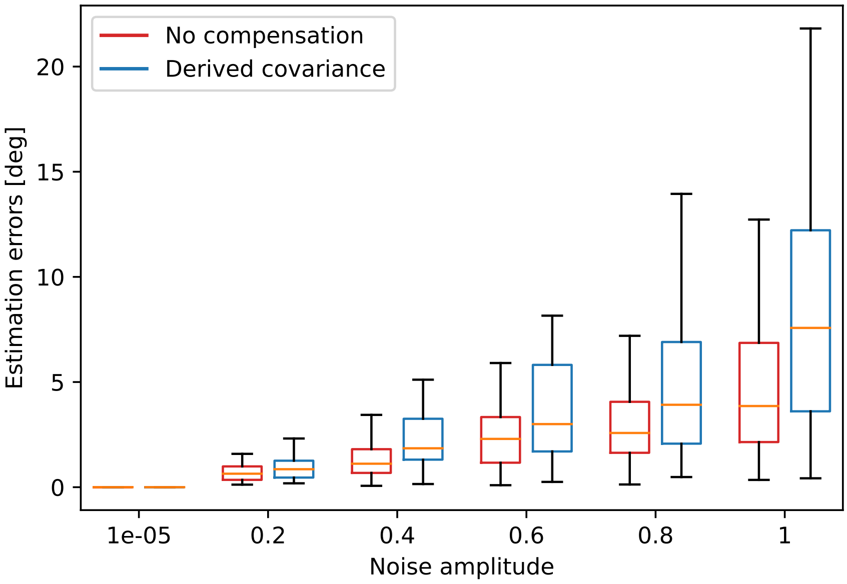


Figure 4.4: Boxplot of estimation errors from using the standard Park-Martin residual (red, left) versus the Mahalonobis norm of Park-Martin residuals with the previously derived residual covariance (blue, right). Note that outliers are not plotted, but that these were at most 10° larger than the outermost error.

ditionally, during derivation of the covariance in Section 3.2 it was noted that its expression requires knowledge of the *true*, noiseless ship and sensor rotations, $\bar{\mathbf{R}}_{\mathrm{nb}}(t_j)$, which would not be known in a real application. For this test, the noisy rotations were therefore used instead. The resulting boxplot of measured errors for this test is seen in Figure 4.4.

The results in Figure 4.4 show the covariance compensated residual to never outperform the non-compensated residuals, for the tests performed. This seems to be the case for both the median estimation error (orange) and variance of errors (size of boxes and whiskers), thus the covariance compensated residuals show performance objectively worse than the non-compensated residual. This result was slightly surprising, but as noted in Section 3.2 there are multiple considerations to make when using the derived covariance.

In an effort to challenge the assumptions made during the derivation of residual covariance, the different combinations of the considerations presented in Section 3.2 were tested. The results of which are seen in Figure 4.5. In this figure, the different combinations are denoted as:

- *No compensation* – The standard Park-Martin residual.
- *Naïve cov.* – The residual covariance presented in Section 3.2.1, using only the covariance of the last absolute pose in each relative pose.
- *Derived cov.* – The derived covariance of Park-Martin residual based on co-

variances on absolute rotations.

- *Noisy rots* – Using the noisy ship and sensor rotation matrices.
- *Noiseless rots* – Using the true ship and sensor rotations. Note: These would not be known in any real-world application, and this is only tested as a debugging measure.
- *Est. extr.* – Inserting the estimated extrinsics $\mathbf{R}$ at each optimization step, and thus updating it iteratively.
- *GT extr.* – Inserting the ground-truth extrinsics $\mathbf{R}_X$ at each optimization step. Note that this would not be possible in a real-world application, as $\mathbf{R}_X$ is the object to be estimated.

To generate the results in Figure 4.5, 80 runs of the same dataset with different noise realizations were tested. The *all data relative first* data-selection strategy presented in Section 3.4 was used for simplicity. In Figure 4.5, only the mean error across all runs of a given noise level is plotted for readability.

Note how the error from using the noiseless rotations compared to using noisy rotations seemingly coincide, for any choice of estimated or ground-truth extrinsics. The reverse is however not true, as the error from using estimated extrinsics is seemingly higher than the error resultant when using the ground-truth extrinsics. The proposed covariance methods perform better than the *naïve* method, which reflects expected behavior.

Despite this, all combinations of methods tested in Figure 4.5 seemingly produce worse estimates than simply not taking the covariance into consideration.

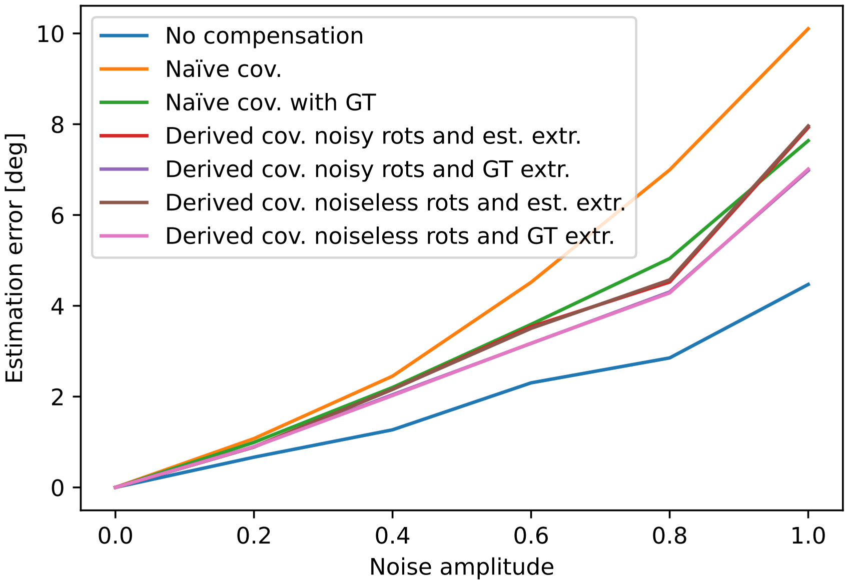


Figure 4.5: Mean estimation errors from 80 runs of different methods of performing covariance compensation

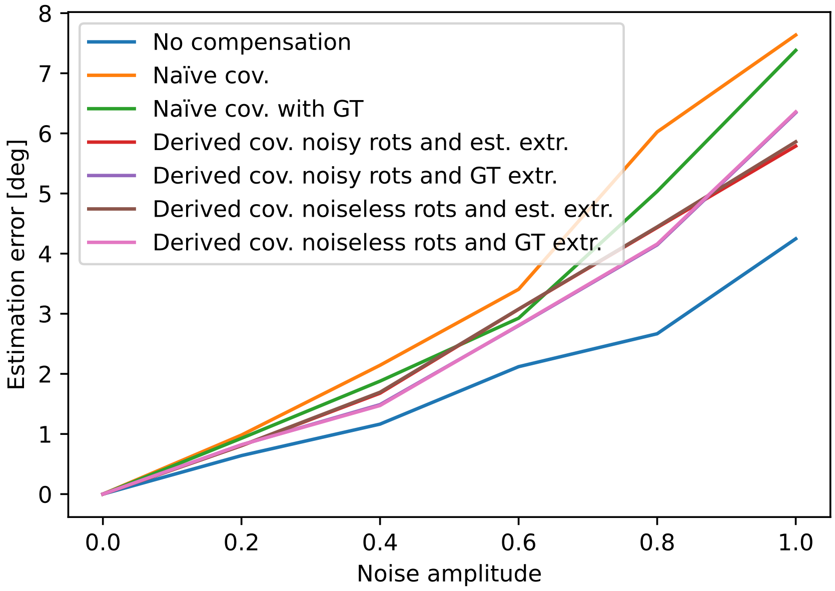


Figure 4.6: Mean estimation errors from 80 runs of different methods of performing covariance compensation when only the latter half of constructed relative poses are used. This ensures their rotation vectors are much larger than the added noise.

Lastly, as mentioned in Section 3.2.3, the assumption of noise being comparably much smaller than movements may not be true for poses close in time. Since the *all data relative first* data-selection strategy was used in these tests, the first couple datapoints are expected to have rotation vectors of possibly very small size. To test for this, Figure 4.6 shows the same simulation setup as in Figure 4.5, but with only the latter half of the constructed relative poses being used for estimation. This ensures that the rotation vectors of the relative poses used have a significant enough size. The resulting performance of the covariance compensation methods is nearly identical to that when using the whole dataset in Figure 4.5.

4.3 Information-weighting

The proposed HE information weights are, as mentioned in Section 3.3.5, an entirely novel metric whose performance must be validated through experiments. The proposed weighting scheme is not unique, and any number of design choices can be implemented differently. Throughout Section 4.3, the *all data relative first* data-selection strategy is used for simplicity, if nothing else is specified.

Firstly, the effect of the SO(3) right Jacobian $\mathbf{J}_r(\boldsymbol{\omega})$ was tested experimentally, as to answer whether ignoring its effect during the derivation of the HE information metric was warranted. This test was done by defining a number of points on the sphere of radius R and a selection of values for the reference rotation vector $\boldsymbol{\omega}$ were chosen. The points on the sphere were transformed by the right Jacobian, along with a set of orthogonal coordinate vectors. The result of this test with $R = 0.3$ and $\boldsymbol{\omega} = [0.5, 0.7, 0.1]$ is seen in Figure 4.7.

Note how none of the points on the sphere are transformed to points of greater distance from the origin. Also, the three orthogonal axes stay orthogonal after the transformation and have the same length. The same result was achieved for different choices of R and $\boldsymbol{\omega}$. Seemingly the right Jacobian only affects the vectors through some rotation.

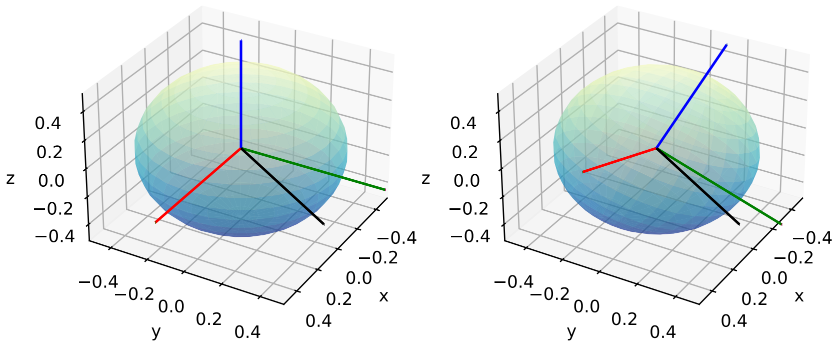


Figure 4.7: Points on the sphere with radius $R = 0.3$ (left) are transformed by the SO(3) right Jacobian, $\mathbf{J}_r(\boldsymbol{\omega})$ (right). The chosen reference rotation vector to the right Jacobian, $\boldsymbol{\omega}$, is seen in black. The red, green and blue vectors are standard coordinate axes, also transformed by the right Jacobian.

Figure 4.8 shows the proposed weighting scheme for the data in the synthetic uniform, as well as for an instance of the synthetic planar dataset. The latter of which has been simulated with a large sudden wave at the timestamp $t = 30$, causing the ship to heel about 30°, as to induce more excitation in the data. This extra excitation is however almost impossible to discern in Figure 4.8, except for a small bump in the middle. The synthetic uniform dataset is seemingly so highly excited, as measured by its information weights being large, that the weights of the synthetic planar dataset are incomparable. The synthetic uniform being much more excited than the synthetic planar dataset is however reflective of intuition regarding the connection between excitation and rotation vectors from Section 2.6.3.

It is not possible to compare the distribution of excitation in the two datsets, since the scale of their weights are so different. To counteract this issue, one could make the design choice to normalize the weights. This is done my defining $w_i' := w_i / \max_i w_i$, which then always would be in the span $0 \leq w_i' \leq 1$. The result of normalizing the weights present in each dataset is seen in Figure 4.9.

In Figure 4.9 it is now possible to discern how the weights of the synthetic

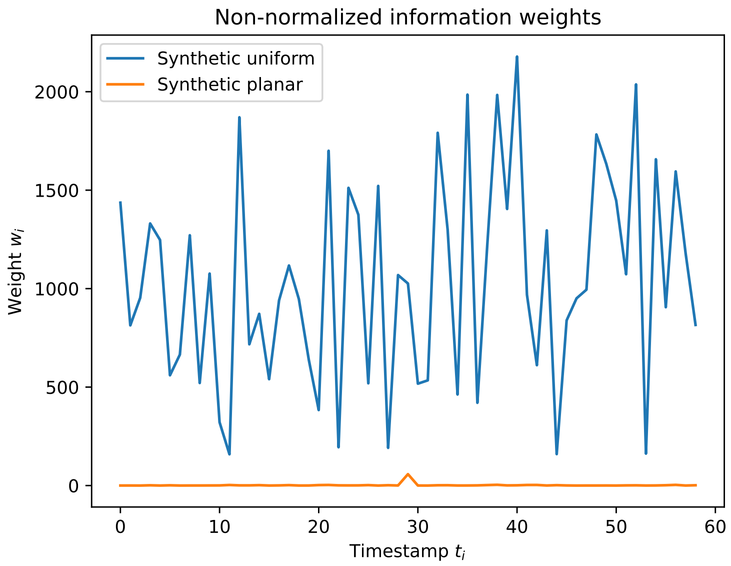


Figure 4.8: The proposed information weighting scheme for each datapoint in the synthetic uniform and synthetic planar datasets. The planar dataset experiences a large wave at timestamp $t = 30$, as to induce more excitation in the estimation problem. Notice that despite this, the weights are much larger overall for the synthetic uniform dataset.

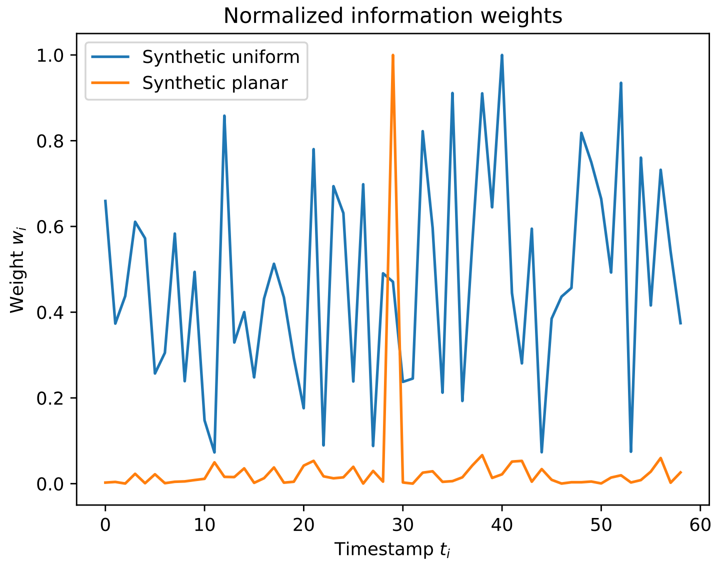


Figure 4.9: The proposed information weighting scheme with normalization, for each datapoint in the synthetic uniform and synthetic planar datasets. The planar dataset experiences a large wave at timestamp $t = 30$, as to induce more excitation in the estimation problem.

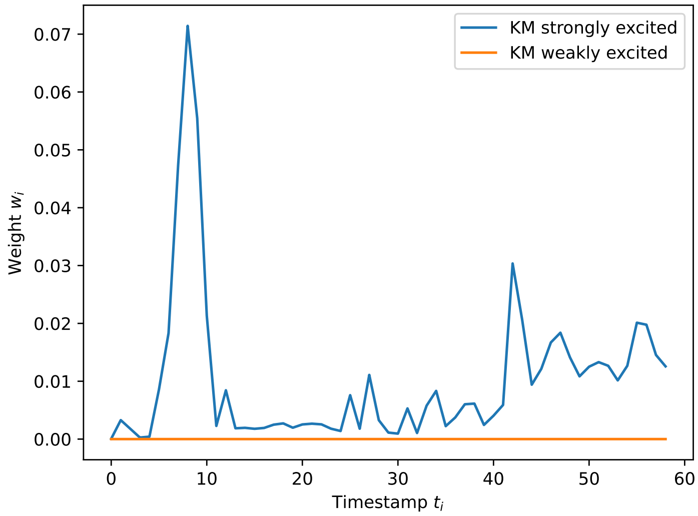


Figure 4.10: The proposed information weighting scheme for each datapoint in the *KM weakly excited* and *KM strongly excited* real-world datasets. Note that even though the weights of the *KM weakly excited* dataset are seemingly zero, they are in fact on the scale of about 10^{-5}.

planar dataset spike at the middlemost timestamp, correctly reflecting the sudden excitation when the ship is hit with the large wave. The distribution of the weights is not changed when normalized, as shown by the weights of the synthetic uniform dataset having the same shape in both Figure 4.8 and Figure 4.9.

Figure 4.10 shows a comparison of the non-normalized excitation weights for the two real-world datasets, the *weakly excited KM* dataset, and the *strongly excited KM* dataset. The weights in the *strongly* dataset are much higher than that of the weakly excited dataset. The weights then reflect the expected qualitative behavior of these datasets in regard to excitation.

With the information metric defined and seemingly operating as expected, more complex questions regarding the Hand-Eye calibration problem can be attempted answered. One such question is the effect of more movement on the excitation. That is, if the ship can be commanded to perform larger turns or sudden stops in an attempt to generate more non-planar rotation vectors, how much more value should a motion which results in 15° pitching be given as compared to a motion which only results in 5° of pitching?

Figure 4.11 shows the distribution of HE information weights and deviations away from purely planar rotation for different wave magnitudes. The results were generated using an instance of the synthetic planar dataset with process noise and no measurement noise, where the wave magnitude A is the scaling of Gaussian wave noise. The dataset was for this test generated *without* the sudden excitation from a large wave. Deviation from planar rotation here means the angle between

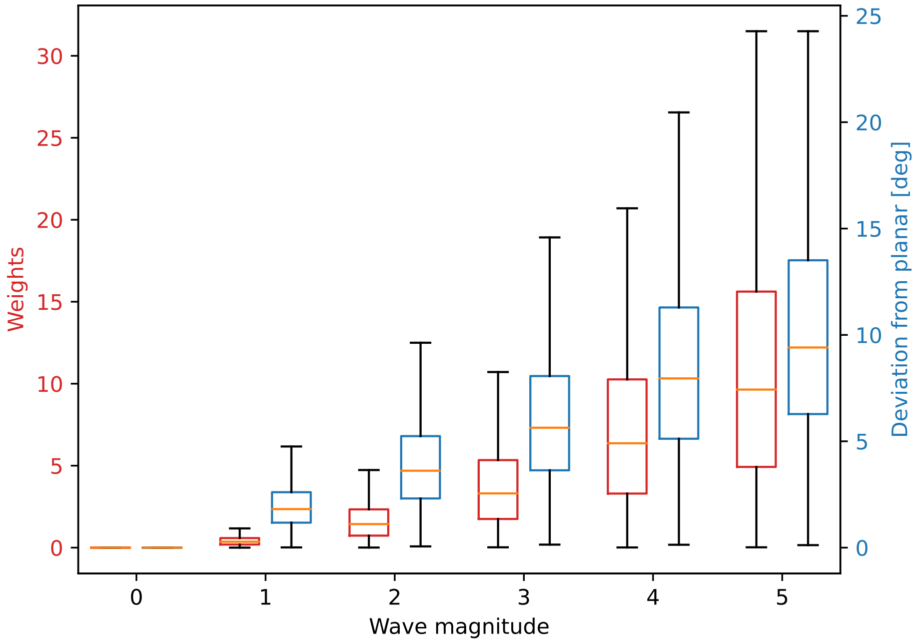


Figure 4.11: The distribution of information weights (red, left) for a planar dataset with variable process noise due to wave motions, resulting in higher deviation away from purely planar rotation (blue, right)

the rotation vector of the ship and the NED Z-axis. Figure 4.11 is the distribution of weights and deviations from the data of 80 different datasets with the given wave amplitude.

Figure 4.11 seemingly shows that increasing wave magnitude leads to increasing weights and deviations, both in terms of their span and median (orange) increasing.

Notice how the medians of the weights and deviations in Figure 4.11 seemingly follow quadratic and linear trends, respectively. This relationship can be investigated further by plotting the weights against the same datapoints' deviation from planar rotation. This plot is seen in Figure 4.12. The resultant relationship is seemingly bounded below by a quadratic function in the deviation. Further, the mean of distributed weights also seems to follow a quadratic line, with decreasing density away from the mean. It should be noted that unlike how the plot is implicitly grouped by dataset through being grouped by wave amplitude in Figure 4.11, the points in Figure 4.12 are taken from all the simulated datasets simultaneously. Thus the distribution of weights/deviations within a single dataset is lost. This is important since the weight of a highly excited datapoint in a dataset with only planar motion is expected to be much higher than the weight of the exact same datapoint in a highly excited dataset, and as such the weight is only relevant when

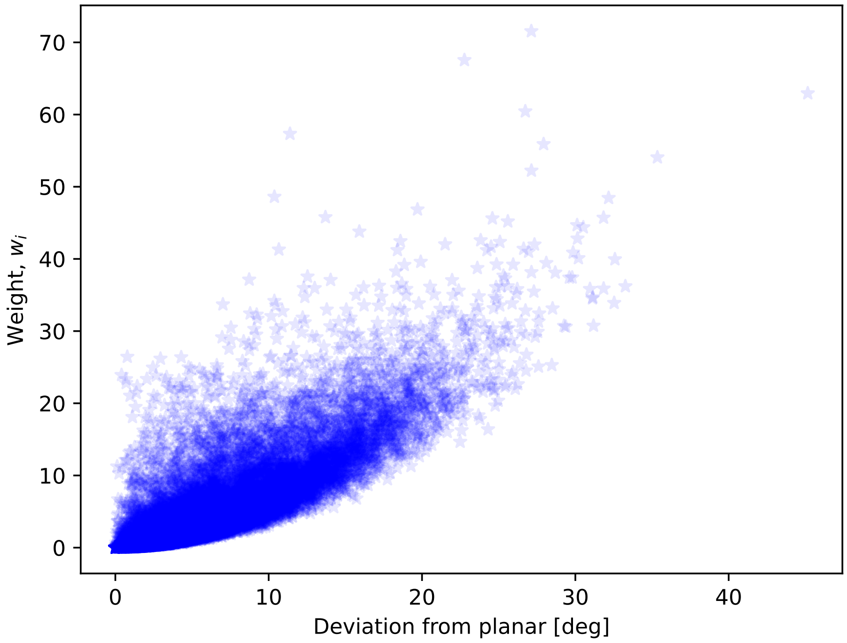


Figure 4.12: Cross plot of weights against deviations calculated in Figure 4.11. Each pair of deviation and weight are plotted with 10% opacity, to discern when many similar pairs are plotted on top of each other. Note the seemingly quadratic lower bound on the plotted points.

compared against the whole dataset.

The results so far have regarded improving the existing understanding of what excitation is in the context of Hand-Eye calibration. For the last test in this chapter we remind ourselves that the main goal of HE calibration is estimating the actual camera orientation, and as such the most important metric of success is the estimation error.

Previous experiences and experiments suggest that when no noise is present, the estimation errors are so small that differences between datasets just as easily can be numerical inaccuracies. On the other hand, if the excitation due to waves is non-existent then estimation errors are so large that differences just as well may be from different choices of initial condition to the optimization. Between these two extremes are many combinations of noise-level and wave-excitation which all may be relevant for any given physical setup. Motivated by this, it is of interest to find a more concrete relationship between how the excitation and noise affect estimates, enabled by the analyses of excitation performed thus far in this thesis.

By associating the distribution of excitation weights with the magnitude of waves in Figure 4.11, a connection between these distributions and estimation error for different noise levels can be made. This is relevant, as the weights alone carry no information, but a distribution of weights does, and associating such distributions with the excitation due to wave motions enable a short-hand description of the excitation for this simulation. Further, while the level of noise present in the measurement of some system is determined by the hardware and as such is not controllable once the ship is at sea, the apparent wave motion that a small to medium sized ship undergoes is somewhat controllable by allowing more exciting motions to be performed. As such, the wave amplitude is a reflection of the required input to lower the error a given amount for some given noise level.

In Figure 4.13, the average estimation error for different combinations of noise and wave excitation is plotted. The averages were calculated over 20 different runs on 10 different instances of the synthetic planar dataset. These datasets were generated without the previously presented inclusion of a sudden large wave, and as such only nominal wave motions induced excitation in the system. For estimating the orientation, the standard unweighted Park-Martin cost-function was minimized.

Figure 4.13 illustrates the relationship mentioned above: Too little wave amplitude or too high noise invariantly leads to unreasonable high estimation errors of around 100° (yellow) for this system. Further, the result suggests a linear relationship between any single level of wave amplitude and the error for different noise levels. Note however how this relationship is seemingly non-linear *between* multiple wave amplitudes, where the lower half of Figure 4.13 has more and more proportion of “very bad estimation results” as opposed to the upper half.

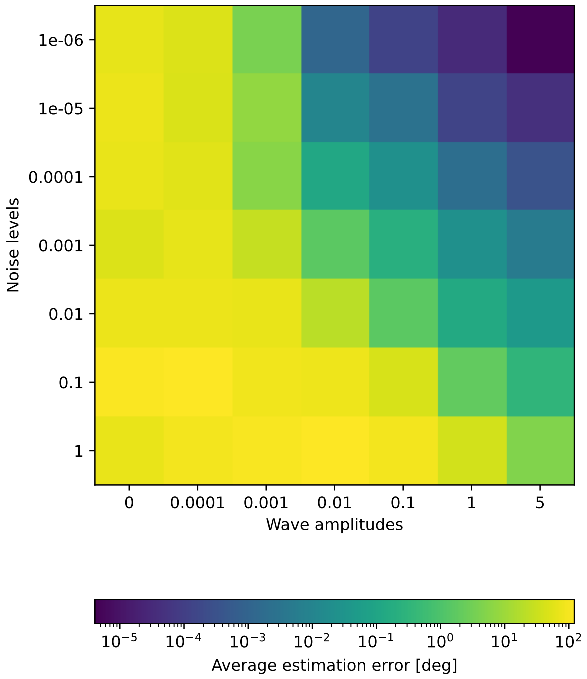


Figure 4.13: Heat map of resulting estimation error for datasets with different combinations of level of measurement noise and excitation due to wave motions. Note the logarithmic scale on both the estimation errors, as well as wave and noise levels.

4.4 Data-selection strategies

For comparison of the data selection strategies presented in Section 3.4, we first take a look at some simple properties.

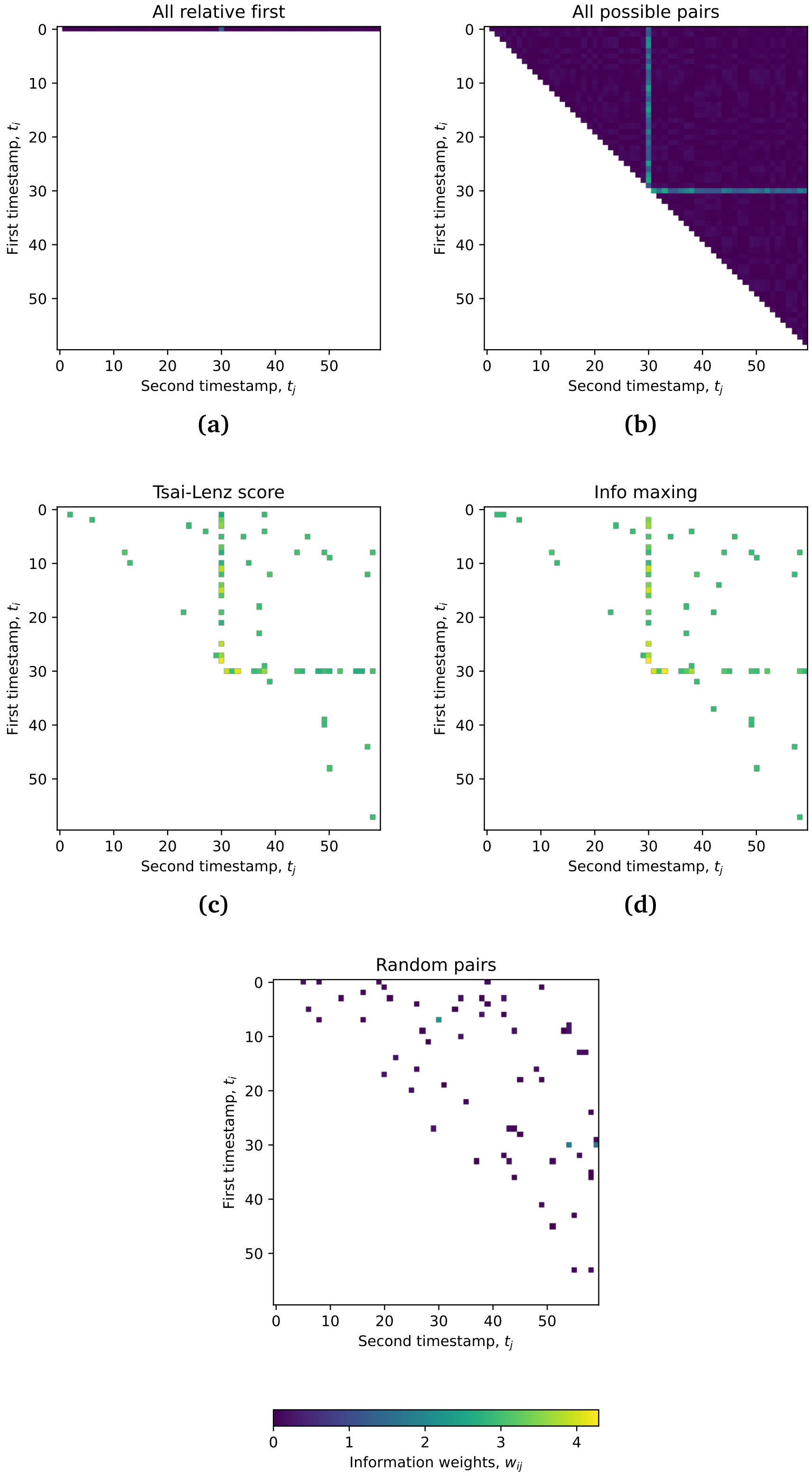
Figure 4.14 shows the chosen datapairs and associated information weights for all presented data-selection strategies over an instance of the synthetic planar dataset with the added sudden wave, as presented in Section 4.1.2. The HE information weights were compensated by the amount of chosen datapoints, as explained shortly in Section 3.3.5, enabling the comparison of the information weights across constructed sets of relative poses of different sizes. This is done by defining $\hat{w}_i = w_i/(M-1)$, where M is the amount of weights produced by the given strategy. Performing this compensation is especially pertinent when comparing any strategy with the *all possible pairs strategy*, which will construct the maximum number of possible pairs.

All methods compared in Figure 4.14 were required to have the second absolute pose making up a single relative pose to have timestamp strictly earlier than the first absolute pose. As explained in Section 2.6.3, this can be performed without loss of generality. The effect of which is the upper-triangular shape of all plots in Figure 4.14.

From the figure, one can verify that the data-selection strategies work as proposed. The *all data relative first* strategy in Figure 4.14a has only weights associated with the 0-th row, while *all possible pairs* in Figure 4.14b has weights associated with every single point on the upper-triangular. The two greedy strategies in Figures 4.14c and 4.14d appear extremely similar both in chosen datapairs and resulting weights, suggesting their maximizing criteria are closely related. The *random pairs* strategy also seems to be working properly, as good as can be evaluated from a singular run of the method.

The underlying dataset has, as mentioned earlier, an added larger wave at $t = 30$ resulting in the ship heeling about 30° away from a planar orientation. Inspecting the figures around the $t = 30$ datapoint, multiple observations can be made. Firstly, the two information-maxing strategies in Figures 4.14c and 4.14d feature the datapoint of large wave prominently, but not exclusively. Both methods also pick a handful of other datapairs as well. Secondly, the weights along $t = 30$ for the dataset constructed by the all possible pairs strategy in Figure 4.14b have seemingly lower magnitude than the weights for the same datapairs in Figures 4.14c and 4.14d. This may suggest a “sunk cost”-type behavior, where having more and more datapoints will in fact decrease the average information in any single datapoair.

Further, it is of interest to compare the performance of the data-selection strategies when it comes to selecting data for use in estimation. This was done by generating 40 different realizations of a noisy synthetic planar dataset and performing Hand-Eye calibration by both minimizing the standard Park-Martin residual and the proposed information-weighted Park-Martin residual for 40 different initial values to the nonlinear optimization procedure. All the data-selection



(a)

(b)

(c)

(d)

(e)

Figure 4.14: Chosen datapairs and associated dataset size-compensated weights for five tested data-selection strategies over the synthetic planar dataset with added sudden wave at $t = 30$. In the figures, white is used to represent a datapair which is not chosen by the relevant strategy.

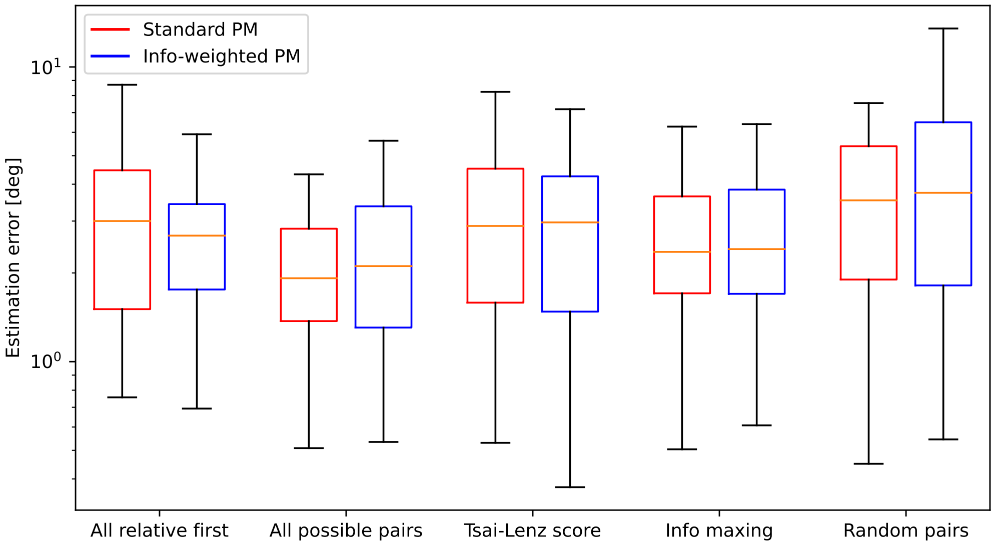


Figure 4.15: Illustration of span in estimation error between the presented data-selection strategies using both the standard Park-Martin residual (left, red) and the proposed information-weighted Park-Martin residual (right, blue). The results were generated with many different instances of noisy synthetic planar datasets. Note the logarithmic scale on the Y-axis.

strategies to be tested, except the *all possible pairs* strategy, were given the same amount of output-datapoints to generate: 60. The resultant boxplot of estimation errors is seen in Figure 4.15.

From Figure 4.15 one can see that all the proposed methods perform about equally as well. All methods produce median estimation errors within 10° error, with the best median performance attributed to the *all possible pairs* strategy. In Section 3.4.1 it was theorized that including as much data as possible by using the *all possible pairs* strategy would lead to the best results when the residuals are properly weighted by their expected information content. This hypothesis then was partially correct, but the info-weighted Park-Martin residual did not improve estimates. In fact, only the *all data relative first* and *Tsai-Lenz score maxing* strategies produced lower estimation errors when the info-weighted Park-Martin residual was optimized. The latter combination did however produce the lowest estimation error of all tested combinations, with an error of about 0.4°. Despite the similarities between the *Tsai-Lenz score maxing* and *info maxing* noted in Figure 4.14, the latter had both lower median estimation error and lower span of errors than the former. Luckily for the integrity of these results, the *random pairs* strategy had the highest median estimation error.

The results in Figure 4.15 suggest the presented data-selection strategies to perform very similarly when estimating over a dataset of 60 datapoints. One of the motivating factors for investigating such strategies was the possibility of including fewer datapoints than every single available datapoint, with the thought that this can be beneficial in real-time applications where storage and computation time is limited and the amount of available data is large. Further, if the expected effect on the estimation error of including *one more* datapoint is quantified, then more complex choices regarding the trade-off between dataset size and certainty of estimates can be made.

To test this, the same strategies in Figure 4.15 except *all possible pairs* were tested with a limitation on the constructed dataset size. The *all possible pairs* strategy was excluded from this analysis since reducing its output size goes against the point of having it as a benchmark, and because its working principle goes against the motivation for this test. As a benchmark strategy, the *all data relative first* strategy and *random pairs* are used instead. For the *all data relative first* strategy, the limitation on dataset size was implemented by only using every n-th datapoint of the constructed dataset, where n was the number that would result in a dataset of the demanded size.

The methods were compared on the same datasets as in Figure 4.15, and by measuring the performance in the same way as previously. This means the single, highly-excited dataset associated with added wave motion is included as well. The datasets consisted of 60 absolute pose-pairs.

Figure 4.16 shows the average estimation error of the presented data-selection strategies for different enforced dataset-sizes. Figure 4.16 shows the results by only using the information weighted Park-Martin residual. The results from using the non-weighted Park-Martin were similar enough to be omitted in this section, but are given in Figure B.1 in Appendix B for completeness.

Of note, the *Tsai-Lenz score maxing* and *info maxing* strategies result in low (about 3°) error for as few as 10 chosen datapoints. The two methods' results are also notably similar, once more suggesting as in Figure 4.14 that their maximizing criteria are similar.

The *all data relative first* strategy seemingly resulted in a suddenly decreasing error for some critical number of datapoints. This can be explained by the way its size was limited, as beyond the subset size of 30 the strategy would begin to include the highly excited datapoint at $t = 30$.

The plotted results in Figure 4.16 seemingly also suggest the performance of the strategies to converge as the subset size increases. At a subset size of 60, the performance of simply picking random pairs has almost caught up with the "intelligent" data-selection strategies.

The last test performed to be presented in this section is the comparison of different data-selection strategies on the estimation error on a real-world dataset. For this, the *KM weakly excited* dataset was chosen as input, since estimation on the *strongly excited* were shown in the specialization project to be slow and difficult, due to much higher amounts of noise than the *weakly excited* dataset. Further, testing on the very weakly excited dataset allows for testing the hypothesis that

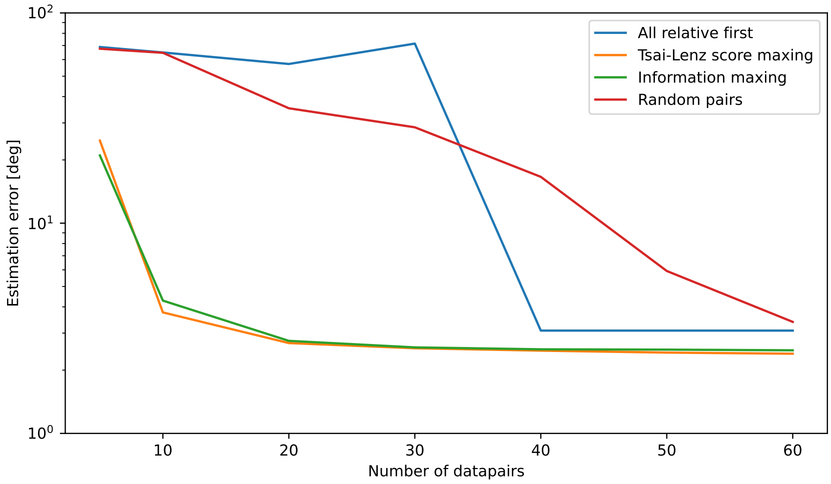


Figure 4.16: Simulation result comparing estimation error from different data-selection strategies and with varying amounts of datapairs. Note the logarithmic scale on the Y-axis.

choosing datapairs intelligently should allow for lower estimation error. The test was performed identically to that seen in Figure 4.15, and the results are seen in Figure 4.17.

Contrary to the results on the synthetic planar dataset in Figure 4.15, the results in Figure 4.17 are wildly different between optimization with the standard Park-Martin residual and info-weighted Park-Martin. Surprisingly though, the *all possible pairs* method performed very well with both optimization techniques. Its median error of about 1.3° is also the lowest estimation error achieved so far, both in this thesis and the preceding specialization project.

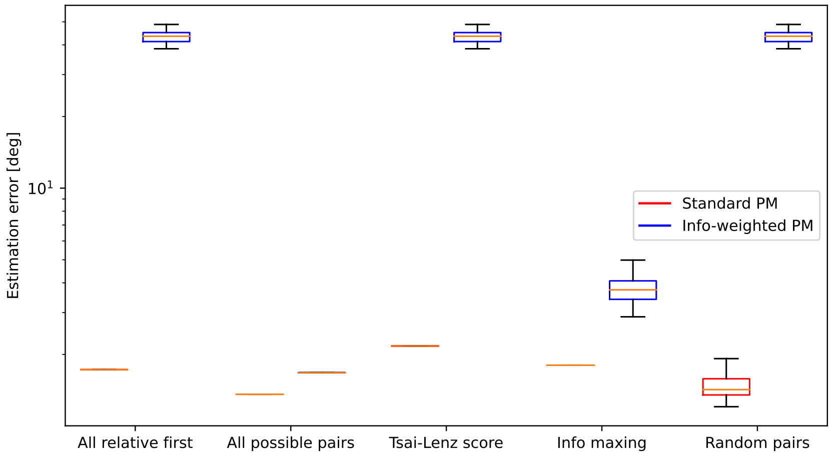


Figure 4.17: Illustration of span in estimation error between the presented data-selection strategies using both the standard Park-Martin residual (left, red) and the proposed information-weighted Park-Martin residual (right, blue). The results were generated using the *KM weakly excited* real-world dataset as input. Note the logarithmic scale on the Y-axis.

Chapter 5

Discussion

5.1 The Park-Martin cost-function and its properties

The derivation of a closed-form expression of the level sets of the Park-Martin cost-function, and associated properties, are, in the author's eyes, exciting. This however does not make them useful, practical or even any good at all. Whereas the derivations of the approximate Hessian and consequent information metric have been used to improve estimates of the Hand-Eye pipeline for ship-data, the closed-form level sets have not been applied in practice further in this thesis.

Some possible use-cases for the derived closed-form expression is to optimize along the level sets. By using the closed-form expression of the solutions and iteratively approach the level $C = 0$, knowledge of the level sets could potentially be used to improve convergence or other properties of the solvers.

The geometric derivation of the level-set parametrization also allowed for an intuitive assertion of an upper bound on the cost-function. Interestingly, a similar upper bound on the approximate Hessian was found in Equation (3.39) of Section 3.3.3. One can imagine being able to combine these two results to obtain a tighter bound on the error of the approximate Hessian.

Lastly, analysis of residuals on the form $\mathbf{r}_i(\mathbf{R}) = \boldsymbol{\alpha}_i - \mathbf{R}\boldsymbol{\beta}_i$ is not restricted to analysis of the Hand-Eye calibration problem. In fact, this is exactly the same function to be minimized when aligning sets of point clouds, for example in the field of Computer Vision [26, 33]. Study of this connection was not in the scope of this thesis, but the similarities are enthralling.

5.2 Using knowledge of the covariances

In this thesis, the covariance of the Park-Martin residual based on covariances of the absolute orientation measurements was derived. This was done following fairly new theory in robotics regarding how noise over group elements should be described [14–16]. Despite intuition saying that compensating for the uncertainty of individual measurements should always result in better estimates, we were not able to show the derived covariance exhibiting this in simulations.

Any number of reasons could be the cause of this discrepancy between expected behaviour and simulated results.

Firstly: There could be an error, either logic or algebraic, in the derivation of relative poses' covariances done in this work. This is difficult to debug in any other way than simply going over the derivations multiple times, which was attempted. No outright error has been found.

Secondly, a number of possible sources of errors were noted in Section 3.2.3. Most of these were tested for, but possibly not well enough. Additionally the noted challenge of the BCH approximation being used twice during derivations was not tested for.

Lastly, there could be an error in the implementation of the Mahalonobis norm or otherwise in how the residuals are being compensated. To test for this, the software was debugged by running the same simulations as presented in Section 4.2, but by adding noise directly to the noiseless rotation vector datapairs $(\boldsymbol{\alpha}_i, \boldsymbol{\beta}_i)$. This did lead to lower errors for *some* datasets, but not consistently. As such it is difficult to conclusively say whether this effect is due to error in the software, erroneous theory, or both.

If some inferences are to be drawn from the simulation results in Section 4.2, they are as follows. It seems like there is no problem in using the noisy rotations, as the plots from using noisy rotations coincide with the plots of noiseless rotations. Additionally, the simulations suggest that using the estimated camera-orientation always results in higher errors than using the ground-truth. The *naïve* covariance was consistently outperformed by the proposed covariance, pointing to the latter at least being more on the right track than the former. However, not much weight should be put on these inferences, as they all stem from the same assumed erroneous results. As such, no real conclusions on the topic of covariance compensated Hand-Eye calibration can be drawn with any certainty.

5.3 The proposed information weighting

The proposed measure of information present in a dataset for Hand-Eye calibration has been shown to be distributed in a way that nicely reflects the expected qualitative behaviours of tested datasets. The weights associated with the highly excited *synthetic uniform* dataset resulted in weights about 1000 times the magnitude of the *synthetic planar* dataset, as shown in Figure 4.8. This was also the case of the real-world datasets in Figure 4.10 where the information weights correctly identify the strongly excited dataset from the weakly excited dataset. The magnitude of the proposed weights is then successful as a method of comparing the magnitude of excitation present in different datasets. However, this comparison also highlights a weakness with the proposed weighting scheme. As mentioned in Section 4.1.2, the *KM strongly excited* dataset was in previous work shown to result in a very noisy SfM reconstruction of the camera poses, and the weighting scheme is calculated as the rotation vector of exactly the reconstructed camera rotations. As such, the weighting scheme will additionally pick up this noise, and it has no method of discerning high amounts of excitation from high amounts of noise. This has the potential to severely impact performance, as it would imply the proposed *info maxing* data-selection strategy to prefer picking the noisiest datapairs out of a noisy dataset, as well as the weighted Park-Martin optimization to assign highest importance to the most noisy datapairs. This latter point could be the reason for the poor performance when using info-weighted Park-Martin in Figure 4.17. This fault could be attempted alleviated by instead calculating the weights based on relative ship movements, but this must only be done if the ship pose measurements are expected to be less noisy than camera pose estimates. Another point is that for any given dataset, the level of noise on measurements and expected excitation can come in a wide range of combinations. In this thesis, both the "somewhat noisy but very planar" *KM weakly excited* dataset and the "very noisy but also highly excited" *KM strongly excited* datasets were both relevant for testing. Therefore the expected usefulness of the proposed information metric is dependent on the ratio between excitation of the system and present noise levels. This is also reflected in Figure 4.13 where this relationship leads to a seemingly non-linear relationship with the estimation error.

The fact that different datasets may produce weights, and thereby costs, of totally different order of magnitude under the proposed weighting scheme also poses a challenge to the usefulness of said scheme, as this can cause the same optimization method to perform wildly differently on the two datasets. This would make weighting less robust for a real-world application where the spectrum of excitation in the dataset may change many times during the lifetime of a real-time use of HE. It is not desirable to have to tune parameters of the optimization method for every qualitative change in the dataset. This is compensated for by normalizing the weights, and we saw in Figures 4.8 and 4.9 how the normalization of the information weights allowed for comparing the distribution of excitation across datasets of different magnitude of excitation. The practicality of this was shown by example in Figure 4.9, wherein we could conclude that the synthetic uniform dataset not only has higher excitation in terms of magnitude, but its data is consistently highly excited. This was a counter to the synthetic planar dataset, whose points of high excitation are centered exclusively around the timestamp of the sudden wave motion. These two characteristic cases led to two very different levels of estimation error, as shown in earlier work by the author given in Appendix C. This could be investigated further by somehow quantifying different datasets' excitation by the variance of their information weights' distribution, and comparing this variance against obtained estimation errors. It would be interesting to see how the estimation errors evolve from the very narrow excitation distribution of the synthetic planar dataset, to the wide distribution of excitation present in the synthetic uniform dataset. In that regard, uniformly distributed poses can be seen as the limit of planar motion with more and more wave excitation. Performing this normalization did however remove information on the magnitude of excitation, which also has an effect on estimation errors. Then one could attempt characterizing datasets by both their mean and variance of their calculated HE information metrics, or by plotting their histograms, not unlike previous works by the author as well as [7].

An alternate method of transforming the generated information metric to more easily compare different datasets' excitation was by compensating by the size of the generated dataset. We saw in Section 3.3.5 how the weights for a dataset of size M is proportional with $M-1$. By normalizing weights by this size, we saw in Figure 4.14 how the weights of both large and small chosen dataset sizes could be compared and give reasonable results. This comparison was enabled by the information metric being based in theoretical derivations, as opposed to being entirely novel where such a relationship not necessarily is apparent.

The hypothesis that the proposed HE info metric is unable to differentiate between noise and excitation seems to be supported by the results in Figure 4.17. Here, performing info-weighted optimization on the known to be noisy *KM weakly* dataset led to associated estimation errors being much higher than performing non-weighted optimization. Combine this fact with the fact that the information-weighted Park-Martin residual did not result in lower estimation error than the unweighted residuals when tested on the synthetic datasets, as seen in Figure 4.15, then these weaknesses suggest the proposed HE information metric is most useful as a measurement of excitation across different datasets. For future work, it would be of value to decouple the excitation due to noise and excitation due to actual ship movement somehow, especially if estimates on the level of noise are available.

So in summary, results from this study suggest the proposed information metric to be useful for quantifying excitation present in a dataset, but not for performing information-weighted estimation. The proposed measure of excitation enables numerical description and comparison of the qualitative properties regarding observability in Section 2.6.3. It also seems that its usefulness as decision support and data-selection criteria is best when the weights are compensated by dataset size. A possible use-case for the metric, as supported by the results in Figure 4.16, is the ability to quantify whether a given dataset contains enough information to sufficiently estimate the extrinsic parameters to the desired accuracy. One can imagine a use-case of the proposed metric where data is collected, the datasets' information metric is calculated and estimation is only performed if the present information is above some user-defined threshold.

The information metric also enables a discussion on the connection between excitation, movement of the ship and resulting estimation errors. For any actual ship, the variable one can influence is the movement of the ship. One can imagine performing sudden stops and violent turning of the ship to generate more deviation of the ship away from purely planar motion. Figures 4.11 and 4.12 illustrate this relationship, and it seems to be quadratic. This should perhaps not come as a large surprise, since the information weights implicitly depend on the angle between rotation vectors in the dataset, which then are trigonometric functions of the deviation away from purely planar motion. By comparing the weight/deviation-relationship in Figure 4.11 with estimation errors in Figure 4.13, some inferences between deviation from planar motion and expected estimation error can be made. If motions can be commanded which generate movement of the ship functionally identical to wave motions resulting in up to 25° deviations from planar motion (equivalent to a wave amplitude of 5), instead of for example a nominal value of 5° deviation, then the resulting improvement on estimation errors is approximately tenfold. Moreso, the relationship is not linear, implying any improved movement of the ship just has better and better estimation errors. Combine this nonlinearity with the fact that for example the error in detected objects in images using the estimated extrinsics also are trigonometric in the error in extrinsics, and the reward from performing larger excitations of the system is compounded.

In Figure 4.7 we saw how three orthogonal vectors stayed orthogonal and oriented right-handedly when multiplied with the right Jacobian. In addition, their lengths did not change under the transformation. From the linearity of matrix multiplication, this result implies this to be the case of any linear combination of the three axes. And since the axes span $\mathbb{R}^3$, then this must also be the case for any vector in $\mathbb{R}^3$. Then if any vector is seemingly only rotated by the Jacobian, its omission is not unwarranted.

5.4 Comparison of data-selection methods

In this thesis, five strategies for performing data-selection have been proposed and investigated. The motivation behind the introduction of such strategies is the fact that for pairing absolute poses into the relative poses used in Hand-Eye calibration, the method of pairing will significantly impact the estimation error, as shown in Figure 2.5. In addition, it was of interest to analyze whether a strategy could be constructed which would give reasonable estimation error for a smaller subset of the entire dataset, by utilizing some intelligent pairing criteria.

Firstly, in Figure 4.14 we saw how the data-selection strategies operated as they should, and the resulting information weights were illustrated. These results further emphasize the importance of datapairing on excitation in the resulting dataset. Figure 4.14 also shows how having too many datapoints will have a diluting-effect, where the more exciting datapoints are not as exciting relative the whole dataset. These observations were in whole enabled by the proposed information metric.

When performed over a dataset of fixed size, all data-selection strategies performed about equally, as seen in Figure 4.15. More importantly however, when constrained to a dataset of smaller size in the simulation shown in Figure 4.16, the propose “intelligent” data-selection strategies converged to a lower-bound estimation error about 3 times as fast as the naïve data-selection strategies. This result is especially impactful when one of the motivations behind introducing a limit on dataset size is the limited storage and computational power available if HE is to be performed on data gathered in real-time. If no such restrictions are present, results in Figure 4.17 on the *KM weakly excited* dataset with inaccurate camera-reconstruction suggests the *all possible pairs* strategy to be optimal. This could possibly be because of the strategy somehow “averaging” out incorrect measurements, since all possible combinations are considered. The fact that the *random pairs* strategy has the lowest lower-limit error in the same figure can suggest the existence of some optimal combination of datapairs, possibly being the pairs that minimize the resulting effect of noise.

It is also of interest how all strategies compared in Figure 4.16 seemingly converge to the same estimation error when the dataset size becomes large enough. This may suggest the existence of some lower bound on the error possible for any given level of noise, as well as possibly explain how similar the results in Figure 4.15 are across the different strategies.

The good convergence of estimation error in Figure 4.16 was achieved with two very similar greedy data-selection strategies: Maximization of our proposed information metric and maximization of an excitation score based on the theory by Tsai and Lenz [6]. This implies the maximizing metrics to be very similar in nature. One advantage of the information metric proposed in this thesis is that it is entirely based on matrix and vector multiplication, which typically is much faster than computing the sine of the angle between vectors. Either way, both strategies produce reasonable estimates for as few as 10-20 datapoints, and highlight the importance of datapairing on efficient Hand-Eye estimates.

5.5 Critiques of the Park-Martin cost-function

This thesis has in large part involved deriving and testing properties of the Park-Martin cost-function for performing Hand-Eye calibration. The assumption on which all this work is built on is that this cost-function is of any value for this purpose, better than the many other alternatives, and that it is appropriate for our specific case of ship-data. This assumption will now be challenged.

Firstly, the cornerstone of the Park-Martin formulation of the Hand-Eye calibration problem is to perform the SO(3)-logarithm to the rotational part of the Hand-Eye equation. This has the advantage explained earlier of the new formulation being directly tied to the measure of observability for the standard Hand-Eye problem, a property heavily utilized in this thesis. The use of the logarithm is however challenging when it comes to the numerics, as it is unstable for input matrices close to the identity. This was noted previously in Section 3.3.5, where this instability negatively affected the derivations of the information metric. Rotations close to the identity will be especially prevalent in the dataset if the chosen data-selection strategy for some reason chooses to pair datapoints close in time. This should however not be the case for the two presented data-selection strategies with best performance as rated in this thesis, those being the *information maxing* and *Tsai-Lenz score maxing* strategies. This is because they both weigh the importance of a datapair by the length of its respective rotation vector, and as such they will tend to not pick relative poses close to the identity.

A second, less tangible, challenge lies in how the derivation of the Park-Martin formulation moves the Hand-Eye problem from SO(3) into the mixed space of rotations and vectors. Meaning, in the original formulation, $\mathbf{R}_\mathrm{A}\mathbf{R}_\mathrm{X} = \mathbf{R}_\mathrm{X}\mathbf{R}_\mathrm{B}$, all present objects are rotations and the equation is entirely contained within the space of SO(3). The Park-Martin formulation of $\boldsymbol{\alpha} = \mathbf{R}_\mathrm{X}\boldsymbol{\beta}$, however, moves the problem into the mixed space of both rotations and vectors. This has proven to complicate matters in this thesis, for example when deriving the Hessian of the Park-Martin cost-function in Section 3.3.1 led to a mix of Jacobians over both SO(3) and regular vector function Jacobians. This challenge is further compounded when the norm is taken of the Park-Martin residuals to construct the cost-function, which is even less compatible with the group structure of SO(3).

Looking past the Park-Martin solver's disadvantages, a big advantage to using Hand-Eye for any system where sensor egomotion can be performed is its simplicity. The calibration problem can be solved using only captured data and requires no extra infrastructure. Further, the derivation in itself makes use of very few assumptions, in principle only assuming the extrinsics to be constant over the dataset used for estimation. Additionally, this assumption can be validated in a simple matter by evaluating the expression $\mathbf{AX} = \mathbf{XB}$, and confirming the error lies within expected values for the given level of noise over the data. As such, the simple and reasonable assumption of staticity can also formulate a criteria for detecting when the sensor mounting loosens and the assumption is broken.

5.6 Further work

The analyses in this thesis of the Park-Martin cost-function for HE calibration have taken a number of different approaches. Both by analysing its level sets and its Hessian, a new understanding of its properties has been built. Still, the author believes there to be more to uncover to this problem. An example not shown with plots in this thesis is the fact that the Park-Martin level sets of level $C = 0$ and $C = 2||\boldsymbol{\alpha}_i||$ are seemingly orthogonal. The reason and importance of this remains unknown, but it is interesting. It could be related to these points being the minima/maxima of the cost function and the positive/negative definiteness of the cost around these points. Further, by performing analyses of the Hessian we have been implicitly analysing the geometry about the minima, and it would be of interest to extend this analysis to any point along the cost surface.

Also of interest would be the properties of the cost-function as it evolves from the minima to the maxima. This could be done by defining the cost surface of $(\omega_x, \omega_y, \omega_z, F(\boldsymbol{\omega})) \in B_\pi(\mathbf{0}) \times \left[0, \sum_{i=1}^{N} 2||\boldsymbol{\alpha}_i||^2\right] \subset \mathbb{R}^4$ and analysing its properties, especially along the w-dimension. Perhaps by starting with a single residual first and then expanding to the full cost.

All these considerations suggest to the author that these questions may be answerable together and simultaneously by viewing the cost-function in some new light, possibly by the field of differential geometry, or even more specifically by information geometry. However, the author is not well-read enough on these topics to make such an assertion for certain, and as such this is left for further work.

This thesis only regarded the estimation of rotational calibration of ship-mounted cameras. A natural evolution of the methods in this work is then to extend them onto the problem of estimating the position of these cameras as well. The method proposed by Andreff *et al.* [5] then seems reasonable to use, as it can estimate the unknown scale of camera movements simultaneous to the position calibration. Prior experience, however, indicates the solver of Andreff *et al.* to be even more sensitive to planar motion, but the techniques employed in this thesis could help alleviate this.

Lastly, the motivation for looking into purely data-driven extrinsic calibration as opposed to the more standard but high-accuracy use of a calibration plate and other framework came from another project where the camera had loosened somewhat mid-trip. The change in extrinsics was not large, but the incident sent a ripple through to every use-case of the camera data, highlighting the importance of good calibration. With the contributions to observability and quantification of data-quality made in this thesis, it is the opinion of the author that using the Hand-Eye calibration problem in an on-line real-time environment for detection and update of the extrinsic parameters is possible. This could possibly be done with a receding horizon over the datapoints and/or by using the proposed information metric on the present dataset as a determiner of whether the current dataset can estimate the extrinsics to the desired precision.

Chapter 6

Conclusion

This thesis has regarded using the Hand-Eye calibration framework to estimate the orientation of ship-mounted cameras. This was done by expanding the work done in the preceding specialization project by addressing the specific challenges for this method when the input data is collected from ships.

Of special interest has been the topic of excitation and information. The qualitative criteria for observability when it comes to planar Hand-Eye data is well-known, but a quantifiable metric of the information present in any single datapoint is – to the author's knowledge – not existent in literature. In this thesis, such a metric was proposed by analysing a quadratic form of the approximate Hessian of the Park-Martin Hand-Eye solver's residual. This metric correctly identifies highly excited datasets from weakly excited ones, and enables comparison of the information present across the different datapoints in a dataset as well as across datasets of different size.

The second main interest of this thesis was the question of how the pairing of absolute poses into relative poses affects the estimation error. Using the proposed information metric as a maximizing criterion, a data-selection strategy was formulated and compared against a number of novel strategies. This proposed strategy allowed for estimating the camera orientation with lower error using fewer datapoints than most other methods tested, shown through simulations. This highlights the importance of Hand-Eye data selection, especially for real-time applications where the amount of data is high.

An effort was also spent on investigating different ways to perform weighted nonlinear optimization of the Hand-Eye problem, with the intent of using knowledge of covariances or the measure of information to weigh the residuals and thereby achieve lower estimation errors. The covariance of the measurements was derived based on the covariance on absolute rotation measurements, using recent developments on the propagation of noise over group-operations. With this covariance derived, the residuals were whitened using the Mahalonobis norm. Separately, the residuals were also attempted weighted using the proposed information metric, with the intent of more strongly weighting the information-rich but uncommon datapoints. The proposed information weighted Park-Martin residual was not shown to perform better than non-weighted residuals in simulations, and neither did the covariance compensated residuals. Some possible causes for these methods performing poorly were discussed.

The thesis has shown the viability of purely data-driven, and thereby automatic, estimation of ship-mounted cameras' orientation by Hand-Eye calibration with camera egomotion. The challenges that arise when using such a method have been addressed, and specifically the question of what constitutes as good data and how to pick it have been given proposed solutions. These proposed solutions enable further research on the topic, and lays another brick on the path to implementing real-time calibration, fault-detection and re-calibration of ship-mounted cameras' orientation.

Bibliography

[1] Y. Shiu and S. Ahmad, ‘Calibration of wrist-mounted robotic sensors by solving homogeneous transform equations of the form AX=XB,’ *IEEE Transactions on Robotics and Automation*, vol. 5, no. 1, pp. 16–29, 1989. DOI: 10.1109/70.88014.

[2] A. Tabb and K. M. A. Yousef, ‘Solving the robot-world hand-eye(s) calibration problem with iterative methods,’ *CoRR*, vol. abs/1907.12425, 2019. arXiv: 1907.12425. [Online]. Available: http://arxiv.org/abs/1907.12425.

[3] S. Ma and Z. Hu, ‘Hand-eye calibration,’ in *Computer Vision: A Reference Guide*, K. Ikeuchi, Ed. Boston, MA: Springer US, 2014, pp. 355–358, ISBN: 978-0-387-31439-6. DOI: 10.1007/978-0-387-31439-6_168. [Online]. Available: https://doi.org/10.1007/978-0-387-31439-6_168.

[4] H. Longuet-Higgins, ‘A computer algorithm for reconstructing a scene from two projections,’ in *Readings in Computer Vision*, M. A. Fischler and O. Firschein, Eds., San Francisco (CA): Morgan Kaufmann, 1987, pp. 61–62, ISBN: 978-0-08-051581-6. DOI: https://doi.org/10.1016/B978-0-08-051581-6.50012-X. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B978008051581650012X.

[5] N. Andreff, R. Horaud and B. Espiau, ‘Robot hand-eye calibration using structure-from-motion,’ *The International Journal of Robotics Research*, vol. 20, no. 3, pp. 228–248, 2001. DOI: 10.1177/02783640122067372. eprint: https://doi.org/10.1177/02783640122067372. [Online]. Available: https://doi.org/10.1177/02783640122067372.

[6] R. Tsai and R. Lenz, ‘A new technique for fully autonomous and efficient 3D robotics hand/eye calibration,’ *IEEE Transactions on Robotics and Automation*, vol. 5, no. 3, pp. 345–358, 1989. DOI: 10.1109/70.34770.

[7] J. Schmidt and H. Niemann, ‘Data selection for hand-eye calibration: A vector quantization approach,’ *The International Journal of Robotics Research*, vol. 27, no. 9, pp. 1027–1053, 2008. DOI: 10.1177/0278364908095172. eprint: https://doi.org/10.1177/0278364908095172. [Online]. Available: https://doi.org/10.1177/0278364908095172.

[8] T. I. Fossen, 'Kinematics,' in *Handbook of Marine Craft Hydrodynamics and Motion Control*. John Wiley & Sons, Ltd, 2011, ch. 2, pp. 15–44, ISBN: 9781119994138. DOI: `https://doi.org/10.1002/9781119994138.ch2`. eprint: `https://onlinelibrary.wiley.com/doi/pdf/10.1002/9781119994138.ch2`. [Online]. Available: `https://onlinelibrary.wiley.com/doi/abs/10.1002/9781119994138.ch2`.

[9] *OpenSfM*. [Online]. Available: `https://opensfm.org/`.

[10] R. Szeliski, *Computer Vision: Algorithms and Applications*, 2nd ed. Springer Cham, 2022. DOI: `10.1007/978-3-030-34372-9`.

[11] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, 2004. DOI: `10.1017/CBO9780511811685`.

[12] J. Solà, J. Deray and D. Atchuthan, 'A micro Lie theory for state estimation in robotics,' *CoRR*, vol. abs/1812.01537, 2018. arXiv: `1812.01537`. [Online]. Available: `http://arxiv.org/abs/1812.01537`.

[13] F. Park and B. Martin, 'Robot sensor calibration: Solving AX=XB on the Euclidean group,' *IEEE Transactions on Robotics and Automation*, vol. 10, no. 5, pp. 717–721, 1994. DOI: `10.1109/70.326576`.

[14] J. G. Mangelson, M. Ghaffari, R. Vasudevan and R. M. Eustice, 'Characterizing the uncertainty of jointly distributed poses in the Lie algebra,' *IEEE Transactions on Robotics*, vol. 36, no. 5, pp. 1371–1388, 2020. DOI: `10.1109/TRO.2020.2994457`.

[15] A. Long, K. Wolfe, M. Mashner and G. Chirikjian, 'The banana distribution is Gaussian: A localization study with exponential coordinates,' in *Robotics: Science and Systems VIII*. 2013, pp. 265–272.

[16] T. D. Barfoot and P. T. Furgale, 'Associating uncertainty with three-dimensional poses for use in estimation problems,' *IEEE Transactions on Robotics*, vol. 30, no. 3, pp. 679–693, 2014. DOI: `10.1109/TRO.2014.2298059`.

[17] D. Q. Huynh, 'Metrics for 3D Rotations: Comparison and Analysis,' en, *Journal of Mathematical Imaging and Vision*, vol. 35, no. 2, pp. 155–164, Oct. 2009, ISSN: 1573-7683. DOI: `10.1007/s10851-009-0161-2`. [Online]. Available: `https://doi.org/10.1007/s10851-009-0161-2` (visited on 29/05/2023).

[18] M. Spong, S. Hutchinson and M. Vidyasagar, *Robot Modeling and Control*, en. Nashville, TN: John Wiley & Sons, 2006.

[19] E. Brekke, *Fundamentals of Sensor Fusion: Target tracking, navigation and SLAM*. Nov. 2020, Unpublished.

[20] J. Nocedal and S. J. Wright, *Numerical Optimization*. Springer New York, 2006. DOI: `10.1007/978-0-387-40065-5`. [Online]. Available: `https://doi.org/10.1007/978-0-387-40065-5`.

[21] N. H. Khan and A. Adnan, 'Ego-motion estimation concepts, algorithms and challenges: An overview,' *Multimedia Tools and Applications*, vol. 76, pp. 16 581–16 603, 2017.

[22] J. Fuentes-Pacheco, J. R. Ascencio and J. M. Rendon-Mancha, 'Visual simultaneous localization and mapping: A survey,' *Artificial Intelligence Review*, vol. 43, pp. 55–81, 2012.

[23] B. Lucas and T. Kanade, 'An iterative image registration technique with an application to stereo vision (IJCAI),' vol. 81, Apr. 1981.

[24] K. Daniilidis, 'Hand-eye calibration using dual quaternions,' *The International Journal of Robotics Research*, vol. 18, no. 3, pp. 286–298, 1999. DOI: 10.1177/02783649922066213.

[25] T. V. Haavardsholm, *A handbook in visual SLAM*, Sep. 2021. [Online]. Available: https://github.com/tussedrotten/vslam-handbook.

[26] J. Wu, Y. Sun, M. Wang and M. Liu, 'Hand-eye calibration: 4-D Procrustes analysis approach,' *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 6, pp. 2966–2981, 2020. DOI: 10.1109/TIM.2019.2930710.

[27] C. Park, P. Moghadam, S. Kim, S. Sridharan and C. Fookes, *Spatiotemporal camera-lidar calibration: A targetless and structureless approach*, 2020. arXiv: 2001.06175 [cs.RO].

[28] F. Dellaert and M. Kaess, *Factor Graphs for Robot Perception*. 2017.

[29] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt and SciPy 1.0 Contributors, 'SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python,' *Nature Methods*, vol. 17, pp. 261–272, 2020. DOI: 10.1038/s41592-019-0686-2.

[30] J. L. Schönberger and J.-M. Frahm, 'Structure-from-motion revisited,' in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[31] R. E. Miles, 'On random rotations in R^3,' *Biometrika*, vol. 52, no. 3/4, pp. 636–639, 1965, ISSN: 00063444. DOI: 10.2307/2333716. [Online]. Available: http://www.jstor.org/stable/2333716 (visited on 11/11/2022).

[32] K. Shoemake, 'III.6 - uniform random rotations,' in *Graphics Gems III (IBM Version)*, D. KIRK, Ed., San Francisco: Morgan Kaufmann, 1992, pp. 124–132, ISBN: 978-0-12-409673-8. DOI: https://doi.org/10.1016/B978-0-08-050755-2.50036-1. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780080507552500361.

[33] T. Viklands, 'Algorithms for the weighted orthogonal Procrustes problem and other least squares problems,' 2006.

Appendix A

Additional mathematical proofs

A.1 The right Jacobian of SO(3) has full rank

Proposition 5. *The right Jacobian of the* SO(3) *group,*

$$\mathbf{J}_r(\theta\mathbf{a}) = \mathbf{I}_{3\times3} - \frac{1-\cos(\theta)}{\theta^2}[\mathbf{a}]_\times + \frac{\theta-\sin(\theta)}{\theta^3}[\mathbf{a}]_\times^2\,, \tag{A.1}$$

has full rank.

Proof. Let us assume $\exists\ \mathbf{v}\in\mathbb{R}^3,\ \mathbf{v}\neq\mathbf{0}$ such that $\mathbf{J}_r(\theta\mathbf{a})\mathbf{v}=\mathbf{0}$. Then

$$\begin{aligned}\mathbf{J}_r(\theta\mathbf{a})\mathbf{v} = \mathbf{v} - \frac{1-\cos(\theta)}{\theta^2}[\mathbf{a}]_\times\mathbf{v} + \frac{\theta-\sin(\theta)}{\theta^3}[\mathbf{a}]_\times^2\mathbf{v} &= \mathbf{0}\\ \implies \frac{\theta-\sin(\theta)}{\theta^3}[\mathbf{a}]_\times^2\mathbf{v} - \frac{1-\cos(\theta)}{\theta^2}[\mathbf{a}]_\times\mathbf{v} &= \mathbf{v}\\ \implies \left(\frac{\theta-\sin(\theta)}{\theta^3}[\mathbf{a}]_\times - \frac{1-\cos(\theta)}{\theta^2}\mathbf{I}_{3\times3}\right)[\mathbf{a}]_\times &= \mathbf{I}_{3\times3}.\end{aligned} \tag{A.2}$$

The last line is exactly the definition of the inverse, and implies

$$[\mathbf{a}]_\times^{-1} = \frac{\theta-\sin(\theta)}{\theta^3}[\mathbf{a}]_\times - \frac{1-\cos(\theta)}{\theta^2}\mathbf{I}_{3\times3}. \tag{A.3}$$

This is a contradiction, as $[\mathbf{a}]_\times$ is not invertible, as proven in the proof of Proposition 3. The skew-symmetric matrix has null-space $\text{null}([\mathbf{a}]_\times) = \text{sp}(\mathbf{a})$ when $\mathbf{a}\neq\mathbf{0}$ and is trivially non-invertible if $\mathbf{a}=\mathbf{0}$. Therefore, by contradiction, $\mathbf{v}$ cannot exist and the right Jacobian is full rank. □

Appendix B

Additional figures

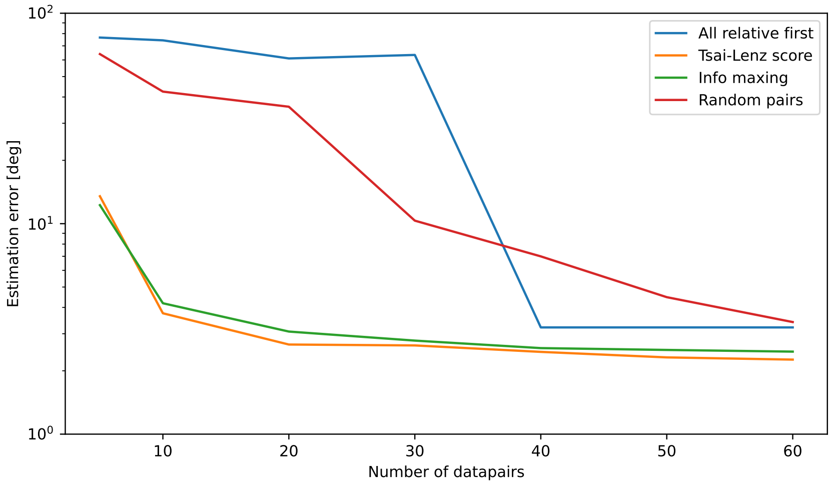


Figure B.1: Simulation result comparing estimation error from different data-selection strategies and with varying amount of datapairs, performed using the standard Park-Martin residual. Note the logarithmic scale on the Y-axis.

Appendix C

FUSION2023 Paper

The following paper was written for- and accepted to the Fusion 2023 conference. The paper was in large part based on the findings of the project performed as preporatory work for the master's thesis at hand. The paper is a preprint and is not attached with a IEEE-citation to the conference proceedings, as these are not published in time for the deadline of this thesis.

Paper attached in full with permission from the author.

- D. Bjerkehagen, E. F. Brekke, E. I. Grøtli and J. Tjønnås, 'Automatic Estimation of Ship-Mounted Cameras' Orientation by Hand-Eye Calibration', *International Conference on Information Fusion (FUSION)*, 2023. Accepted.

Automatic Estimation of Ship-Mounted Cameras' Orientation by Hand-Eye Calibration

Daniel Bjerkehagen*, Edmund Førland Brekke*, Esten Ingar Grøtli†, Johannes Tjønnås†

***Abstract*—By developing a method for automatically calibrating the extrinsic parameters of ship-mounted cameras, this paper tests combining Structure from Motion-algorithms with Hand-Eye calibration solvers in a novel algorithm which demonstrates an ability to discern the orientation of cameras with accuracy comparable to- or better than current manual methods, proven through tests with both synthetic and real-world data.**

***Index Terms*—hand-eye calibration, structure-from-motion, autonomous vessels, extrinsic parameters**

I. Background and motivation

Of the research being done on autonomous systems, the case of autonomous ships has shown to be both worthwhile academically and strongly motivated by the industry. Recent advances in the field of autonomous vehicles has relied on innovations in computer vision to enable use of the dense data cameras offer, with autonomous vessels being no exception. As with all sensors, successfully applying the information provided by cameras requires it to be calibrated within satisfactory precision. One important set of calibration-parameters relevant to most use-cases and which are often estimated by manual measurements are the extrinsic parameters, meaning the relative position and orientation of the camera.

One popular method for performing estimation of the extrinsic parameters for the case when the camera is mounted on a robot arm involves solving the *Hand-Eye calibration problem* [1]. Research has shown some Hand-Eye (HE) solvers able to get estimates of the extrinsic parameters as close as within 0.1° and $2\,\mathrm{mm}$ of the ground truth parameters in optimal controlled experiments [2]. The mathematical properties required of input data to yield the extrinsics observable through solving the Hand-Eye equations has been known for quite some time [1], [3], but authors provide mostly general guidelines rather than optimal strategies for selecting subsets of data in large datasets. Furthermore, most research on the topic relies on the use of geometric calibration targets to reconstruct the camera motion to the correct scale.

The problem of estimating camera motion is of interest in computer vision, and multiple algorithms exist for reconstructing the motion of a camera even when a calibration target is unavailable. Such methods often rely on feature-detection and tracking algorithms, and some have become widely successful due to a number of impressive results such as Visual Localization And Mapping (VSLAM) and Visual Odometry (VO) [4]. The algorithms used often have the disadvantage of such reconstructions not being to metric scale, requiring fusing the reconstruction with auxiliary motion measurements.

Methods exist for combining camera motion estimation with the Hand-Eye calibration formulation, notably the Hand-Eye solver developed by Andreff *et al.* [3], but most other Hand-Eye solvers assume scale of the camera motion is known. For the case of ship-mounted sensors, work has been done on automatically finding the extrinsics of the camera relative a sonar [5]. Roy *et al.* [6] use sensor egomotion reconstruction from a vessel with planar movement to estimate the extrinsic parameters through maximum a posteriori estimation method. To the authors' knowledge, no research exists on the topic of using a camera egomotion reconstruction algorithm as data-baseline for extrinsic calibration of ship-mounted sensors using the Hand-Eye calibration problem formulation.

By identifying the components of the Hand-Eye calibration problem with measurements available when cameras are rigidly mounted on ships, this paper tests a novel algorithm pipeline for estimating the camera extrinsics using only images and measurements of the ship's position and attitude as input to the algorithm. The presented pipeline is thereby purely data-driven. The report also addresses some challenges present when using ship-data for Hand-Eye calibration, and a qualitative way of measuring the excitation in data used for Hand-Eye calibration is presented.

In many applications, including marine operations, the position of the camera is known with high precision due to the accurate construction of ships. This is not necessarily the case for the orientation of the camera. Our work therefore makes a large simplification in only attempting to estimate the orientation of the ship-mounted cameras.

The main contribution and two side-contributions in this paper can be summarized as:

- A novel pipeline for data-driven extrinsic calibration of ship-mounted cameras
- A graphical method for evaluating the excitation present in datasets for Hand-Eye calibration purposes
- To the authors' knowledge, the first paper which use the framework of Hand-Eye calibration to calibrate ship sensors' extrinsics

This research is funded by SFI AutoShip, an 8-year research-based innovation Centre focusing on safe and sustainable autonomous ship operations. We would like to thank our partners, including the Research Council of Norway under Project number 309230. We would especially like to thank Torbjørn Barheim and Kongsberg Maritime (KM) for providing the data set.

*Department of Engineering Cybernetics,
Norwegian University of Science and Technology, Trondheim Norway
daniel.bjerkehagen@gmail.com, edmund.brekke@ntnu.no
†Mathematics and Cybernetics,
SINTEF Digital, Trondheim, Norway
{EstenIngar.Grotli, Johannes.Tjonnas}@sintef.no

II. Egomotion estimation algorithms

Multiple algorithms exist within the field of computer vision which produce an estimate of the 3D motion of a camera relative to some environment, given an ordered set of pictures. Some of the approaches to estimate camera motion include Simultaneous Localization and Mapping (SLAM), Visual Odometry (VO), and Structure from Motion (SfM). These methods are unified in literature under the term "camera egomotion estimation" [4].

Performing estimation of camera egomotion often consists of tracking the movement of notable features in the environment through the pictures for which those features are visible. Assuming these observed features are static relative the environment also allows formulating geometric or numeric equations whose solution is the set of plausible camera motions which caused the observed movement of features. Lastly, most methods refine the initial camera egomotion estimates through softly enforcing some constraint, e.g. the assumed staticity of the environment.

III. The Hand-Eye calibration problem

A. Scaleless Hand-Eye calibration

The Hand-Eye calibration problem originates in robotics, being the issue of finding how a sensor, often a camera, is mounted rigidly relative to an end effector. By analysis of the system certain mathematical properties can be shown and used, something often attributed to be studied first by Shiu *et al.* in 1989 [1]. In the original formulation, the camera and end effector are attached to a robotic arm, allowing for precise movement of the system. The setup is moved between predetermined poses and pictures are taken of a stationary calibration target at each pose. Employing a geometric algorithm, like the 8-point algorithm [7], the camera movement between each picture is recovered. The camera movement may alternatively even be recovered in an unstructured environmnent with no calibration target present by employing an algorithm for camera egomotion estimation. Andreff *et al.* [3] showed that combining the known end effector-movement with such reconstructed camera-movement, the *scaleless* Hand-Eye calibration problem given in Equation (1) can be formulated. Solving this problem for the unknown position and orientation of the camera relative the end effector gives the Hand-Eye calibration.

The equations describing the calibration problem can be written as

$$\mathbf{A}\mathbf{X} = \mathbf{X}\mathbf{B}(\lambda), \tag{1}$$

or equivalently,

$$\begin{bmatrix} \mathbf{R}_\mathrm{A} & \mathbf{t}_\mathrm{A} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_\mathrm{X} & \mathbf{t}_\mathrm{X} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R}_\mathrm{X} & \mathbf{t}_\mathrm{X} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_\mathrm{B} & \mathbf{t}_\mathrm{B}\lambda \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix} \tag{2}$$

These can be divided into rotational and translational parts, that is

$$\mathbf{R}_\mathrm{A}\mathbf{R}_\mathrm{X} = \mathbf{R}_\mathrm{X}\mathbf{R}_\mathrm{B}, \tag{3}$$

$$\mathbf{R}_\mathrm{A}\mathbf{t}_\mathrm{X} + \mathbf{t}_\mathrm{A} = \mathbf{R}_\mathrm{X}\mathbf{t}_\mathrm{B}\lambda + \mathbf{t}_\mathrm{X}. \tag{4}$$

Here, $\mathbf{A}$ is the Homogeneous Transformation (HT) matrix describing the rotation and translation of the end effector between two poses of the system, $\mathbf{B}$ is the HT describing the reconstructed camera motion between two poses and $\mathbf{X}$ is the unknown rigid pose of the camera relative the end effector, also represented as a HT. λ represents the unknown scale of the reconstructed camera translations, since both hand- and eye translations needs to be given in the same unit for the equality to hold. The Homogeneous Transformation matrices are compositions of a rotation matrix and a translation vector like in Equation (2), which enables a mathematical representation of the pose of a frame relative some other frame.

The matrices $\mathbf{A}$ and $\mathbf{B}$ are the so-called *relative poses* of the hand and eye between two *absolute poses*. These terms are to be understood as the following: Poses defined in some inertial frame are said to be absolute poses. If the frame "n" is an inertial frame, while $\mathbf{H}_\mathrm{nb}(t_p)$ and $\mathbf{H}_\mathrm{nb}(t_q)$ are the absolute poses of the end effector at times t_p and t_q, then $\mathbf{H}_\mathrm{b,pq} = \left(\mathbf{H}_\mathrm{nb}(t_p)\right)^{-1}\mathbf{H}_\mathrm{nb}(t_q) := \mathbf{A}_{pq}$ is the relative pose of the end effector between these two absolute poses. The definition is equivalent for the relative pose of the camera, with the camera poses being defined relative some coordinate system defined by the egomotion algorithm. Therefore, from the set of available absolute poses, a set of relative poses must be constructed to be able to employ the Hand-Eye framework for estimating the extrinsics.

B. Hand-Eye formulation for ship-case

The goal in this work is to estimate the orientation of ship-mounted cameras. Modern ships have advanced sensor-suites fusing GNSS measurements with inertial- and attitude measurements, meaning the ship's pose is available with high accuracy. Many modern ships are also equipped with cameras as a part of their sensor rigs.

It is possible to recognize that a ship with such a navigational system and rigidly mounted cameras have all the equipment necessary to formulate a fitting Hand-Eye calibration problem, with the solution being the unknown extrinsics of said cameras. The pose of the ship can be seen as analagous to the pose of the "hand" from the original Hand-Eye formulation, and using an egomotion estimation algorithm on the captured images yields the scaleless "eye" movements. It is therefore conjectured that using the available data, that being the ship's pose and images captured, should allow for data-driven estimation of the cameras' extrinsic parameters.

It has been shown that to be able to uniquely determine all of the extrinsic parameters, that being the Homogeneous Transformation $\mathbf{X}$ in Equation (1), the system must undergo at least two motions with non-parallel axis of rotation [8]. If, however, the system in question undergoes planar translation and with rotation purely about the normal of the plane, the previous condition is consequently not fulfilled. Andreff *et al.* then shows in [3] how two nonzero planar motions still render the sensor orientation and part of the sensor position uniquely observable, but the sensor height relative the plane of motion is not observable.

This last point is especially relevant when it comes to the application of the Hand-Eye calibration problem on ship-data, since ships move mostly in the plane. Actual data generated by ships is not perfectly planar, but comparable to how dividing by a very small - but not exactly zero - number leads to numerical instability, it is expected that estimates generated with ship-data too will be numerically sensitive due to the closeness to degeneracy [8]. This is, as mentioned earlier, only a challenge for estimating the position of a sensor, and two nonzero planar movements is still enough to make the sensor orientation observable.

Another problem when attempting to combine structureless camera egomotion algorithms with the Hand-Eye problem formulation is the missing scale of the camera egomotion reconstruction. Much research is being done on the topic of metric scale estimation, often employing deep learning to achieve this [9]. If no such auxiliary scale estimation is used to correct the reconstructed camera movement, then most methods for solving the Hand-Eye equation will not be able to estimate the positional extrinsic parameters of the camera.

In this work, both the problem of missing scale and planarity are side-stepped by only estimating the orientation of the cameras. This could be further motivated by the fact that for the case of ships, the position of the sensor is often known to high accuracy. Additionally, using a wrongful camera orientation for estimating the position of some object detected by the cameras will typically cause larger error in these estimates than wrongful camera position will, especially when the detected object is far away.

C. Solvers

Many different approaches for solving the Hand-Eye equation have been developed over time, and may generally be categorized into two groups of two: A solver can employ a closed-form or iterative solution, and the solver can either solve for the whole extrinsics simultaneously or step-wise [2]. Closed-form solvers find the theoretically optimal solution given the data and knowledge of the noise model corrupting the data, while iterative solutions are numerical approximations to the optimal solution which can do this without knowledge of the noise, often found through optimization. Simultaneous solution techniques to the Hand-Eye equation find both orientation and position of the sensor at the same time, while step-wise solvers solve for the orientation first and then use that estimate to compute the position. Since only the orientation of the camera is of interest in this work, only the orientation-part of step-wise solvers are tested.

What follows is a short presentation of the Hand-Eye solvers compared in this paper. In the following equations, Log is the function which sends orientations to their respective angle-axis counterparts in the Lie algebra of $\mathrm{SO}(3)$, with notation inspired by Solà *et al.* in [10]. The vectorization function, vec, is the function stacking the columns of a matrix into a large column vector, and $\otimes$ is the Kronecker product. f_i is the ith residual based on the ith $(\mathbf{A}, \mathbf{B})$-data pair, to be used in a nonlinear least squares optimization.

Park and Martin show in [11] that the rotational Hand-Eye equation, Equation (3), can be rewritten as $\mathbf{R}_{\mathrm{X}}\mathrm{Log}\left(\mathbf{R}_{\mathrm{B}}\right) = \mathrm{Log}\left(\mathbf{R}_{\mathrm{A}}\right)$. This equality may then be solved by $\mathbf{R}_{\mathrm{X}} = (\mathbf{M}^{\mathrm{T}}\mathbf{M})^{-1/2}\mathbf{M}^{\mathrm{T}}$, where

$$\mathbf{M} = \sum_i \mathrm{Log}\left(\mathbf{R}_{\mathrm{B},i}\right)\mathrm{Log}\left(\mathbf{R}_{\mathrm{A},i}\right)^{\mathrm{T}}. \tag{5}$$

This solution tactic is hereby dubbed “Park-Martin Closed-form”. Alternatively, one could formulate a nonlinear optimization problem for minimizing the above derived equality, by defining the “Park-Martin residual”

$$f_i(x) = \mathbf{R}(x)\mathrm{Log}(\mathbf{R}_{\mathrm{B},i}) - \mathrm{Log}(\mathbf{R}_{\mathrm{A},i}). \tag{6}$$

Using the formulation by Andreff *et al.* [3], which consists of finding the null-space of a matrix, the residual

$$h_i(x) = (\mathbf{I}_{9\times 9} - \mathbf{R}_{\mathrm{B},i} \otimes \mathbf{R}_{\mathrm{A},i})\mathrm{vec}(\mathbf{R}(x)), \tag{7}$$

can be defined and is named the “AHE residual” after the authors’ initials. Lastly, Park and Martin define in [11] a very simple optimization function, being only the error between the sides of Equation (3) for some given camera orientation $\mathbf{R}(x)$. Using their choice of metric over $\mathrm{SO}(3)$, the residual

$$g_i(x) = \mathrm{Log}\left(\left(\mathbf{R}_{\mathrm{A},i}\mathbf{R}(x)\right)^{\mathrm{T}} \left(\mathbf{R}(x)\mathbf{R}_{\mathrm{B},i}\right) \right) \tag{8}$$

can be defined. This residual is in this paper dubbed the “$\mathrm{SO}(3)$-metric residual”.

D. Graphical evaluation of excitation

As previously stated, Andreff *et al.* showed that the Hand-Eye calibration is observable if two relative poses of non-parallel rotation axes are used for estimation. For the purposes of estimation, using more than the minimal set of data is desirable for generating estimates more robust against noise. Tsai *et al.* describe in [8] how the choice of arm-poses affect the propagation of noise to the estimates of the extrinsic parameters. Their findings are summarized in Equation (9).

$$\mathrm{Var}(\boldsymbol{\omega}_{\mathrm{X}}) \propto \frac{\sqrt{\mathrm{Var}(\boldsymbol{\omega}_{ab})^2 + \mathrm{Var}(\boldsymbol{\omega}_{bc})^2}}{\sin\left[\angle(\boldsymbol{\omega}_{ab}, \boldsymbol{\omega}_{bc})\right]} \sqrt{\frac{1}{||\boldsymbol{\omega}_{ab}||^2} + \frac{1}{||\boldsymbol{\omega}_{bc}||^2}} \tag{9}$$

Here, a, b, c are three different points in time and $\boldsymbol{\omega}_{ab}$ is the rotation axis of the rotation $(\mathbf{R}_{na})^{\mathrm{T}}\mathbf{R}_{nb}$. The formula shows how uncertainty of the data propagates more strongly through to the estimated camera orientation depending on the geometry of the chosen arm-poses. Tsai *et al.* suggest a strategy for how to choose poses of the robot-arm to minimize Equation (9), but this strategy is not applicable in this work since the movements cannot be decided ahead of time by the calibration system. It is desirable to find a data-selection criteria based on Equation (9) which in real-time can select the best datapairs in a stream of high amounts of noisy data.

To this end, this work suggests a first step in developing such a data-selection strategy by providing a graphical method for evaluating the "excitation" present in different datasets. The method evaluates the two factors of Equation (9) which are affected by choice of data. The *type 1* excitation is defined as the size $||\boldsymbol{\omega}_{\mathrm{b}i}||$, with $\boldsymbol{\omega}_{\mathrm{b}i}$ being the rotation vector of relative ship-pose i. *Type 2* excitation is defined as $|\sin[\angle(\boldsymbol{\omega}_{\mathrm{nb}}(t_i), \boldsymbol{\omega}_{\mathrm{nb}}(t_j))]|$, for any two ship-orientations at timestamps t_i and t_j. For both of the defined types of excitation, larger values is better, due to the corresponding factors appearing inverted in Equation (9). The graphical method is presented further in Section V-B.

IV. Implementation details

What follows is an explanation of the parts of the algorithm pipeline, as well as a summary of some questions which challenge the validity of the given output from the algorithm. Some of these questions and design choices are addressed, while some questions are left unanswered as potential further work. An overview of the pipeline is seen in Figure 1.

The positional data from ships' navigational units are often given in a coordinate system preferred by the GNSS-system, like WGS84 or some geodetic coordinate. The Hand-Eye formulation, however, requires translations to be given in inertial Euclidian frames. A first step in the algorithm is to process the positional data to construct a local tangent plane from which to define the NED-coordinate system. For simplicity, the local tangent plane was defined relative the first datapoint, which for the short timespans analyzed in this work was deemed sufficiently accurate.

For performing egomotion-estimation based on captured images, the open-source library COLMAP [12] was used, with the notable alternative of OpenSfM [13] being used during initial testing to verify the COLMAP-reconstructions. The choice to use a SfM-library instead of e.g. VSLAM was again due to simplicity of use. One may imagine using the iterative nature of VSLAM or VO to iteratively improve on an estimate of the Hand-Eye solution.

With the aforementioned components in place, pose-data of the ship is given relative the local tangent plane and the SfM-libraries give camera egomotion relative some arbitrary reference-frame chosen by the library. As explained in Section III-A, the Hand-Eye Problem formulation requires the input $(\mathbf{A}, \mathbf{B})$ pose-pairs be relative poses, that being the change in pose between two points in time instead of relative some absolute reference frame. Since this formulation of the Hand-Eye calibration problem does not allow for arbritary movements of the system, rather the system moves and the algorithm must do best with whats given, a strategy for choosing how the relative poses are constructed must be employed. For this work, the simplest solution of calculating all poses as relative the first pose is employed, but further work should be done on examining the exact effects different methods of constructing relative pose has on the performance of the pipeline.

The algorithm was implemented in Python, using SciPy's [14] nonlinear least squares Gauss-Newton solver for estimating the extrinsics.

V. Simulation results

A. The datasets

For this paper, three datasets are used to generate the results presented. These are the *synthetic uniform*, *synthetic planar* and *real-world* datasets. The synthetic uniform dataset is a set of N poses randomly generated with uniform probability density over $\mathrm{SO}(3)$ and some span of positions, the former being achieved following the method of Shoemake in [15]. A set of extrinsic parameters are chosen arbitrarily, and camera-poses are synthesized by combining the extrinsics with the aformentioned randomly generated poses, creating noise-free knowledge of the camera-poses. Being uniformly distributed, this dataset contains no underlying structure.

The synthetic planar dataset is generated as a random walk on yaw (the rotation angle about the body z-axis) and a simple movement-model. The resulting ship-poses operate mostly in the same plane and with little to no pitch nor roll, leading to noiseless data with similar structure to that of real-world data. Figure 2 shows an example of the generated synthetic ship-poses. The camera-poses are generated in the same way as the camera-poses of the synthetic uniform dataset explained earlier; as the composition of randomly generated ship-poses and chosen ground truth extrinsics.

The real-world dataset was collected from a large passenger cruise ship fitted with camera rigs. The subset of data used for evaluating the algorithm pipeline in this paper were from a sequence spanning 60 seconds when the ship was leaving port. See Figure 3a for an example image captured. This meant a high amount of features could be detected and tracked between each image, but also that the excitation of the ship was minimal. Figure 3b shows the output SfM reconstruction from COLMAP over the real-world dataset. Some other points to note about the real-world dataset is as follows: Firstly, the vessel was fitted with an advanced navigational system, meaning the precision of ship-pose measurements were high. Secondly, it was not known to which numerical value of uncertainty the ground truth extrinsic parameters fit with the actual physical mounting of the cameras. This will pose a challenge when attempting to draw conclusions based on results using this dataset.

B. Visualizing excitation

The following few paragraphs regard figures visualizing the qualitative measure of excitation from Section III-D applied on the three presented datasets. The presence or lack of the two previously defined types of excitation are then evaluated using the graphical measure. Following the explanations in Section III-D, it is expected that datasets which show higher excitation to allow for estimating the extrinsics with lower variance.

Figures 4a and 5a show histograms of $||\boldsymbol{\omega}_i||^2$ over all timestamps i in the synthetic uniform and planar datasets, respectively. This visualization gives an indication of the distribution of type 1 excitation present in these datasets, with more right-shifted values meaning higher excitation. The synthetic

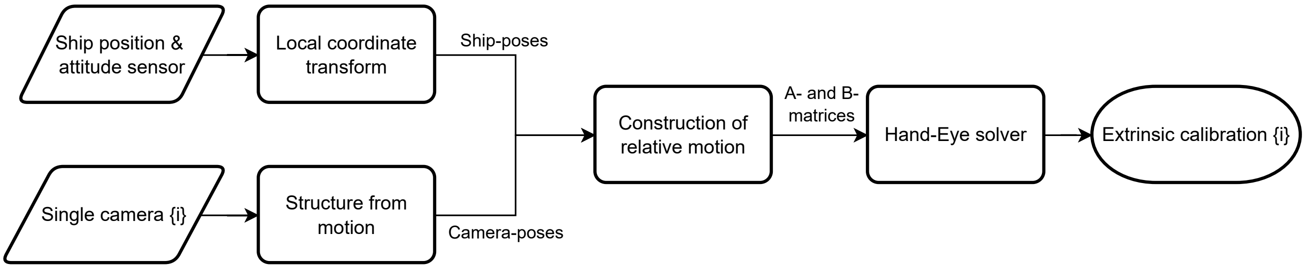

Ship position & attitude sensor
Local coordinate transform
Ship-poses
Single camera {i}
Structure from motion
Camera-poses
Construction of relative motion
A- and B- matrices
Hand-Eye solver
Extrinsic calibration {i}

Fig. 1: Flowchart of the implemented pipeline

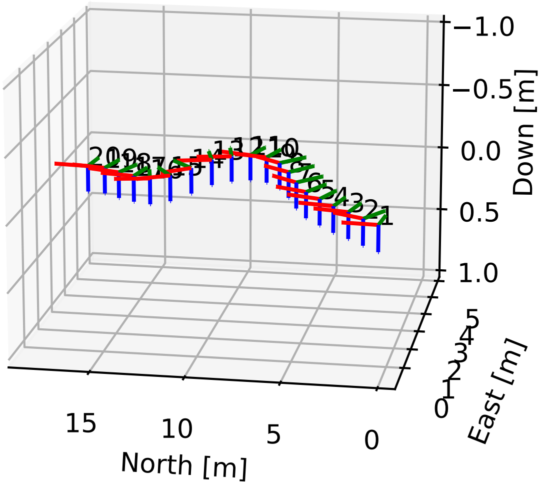
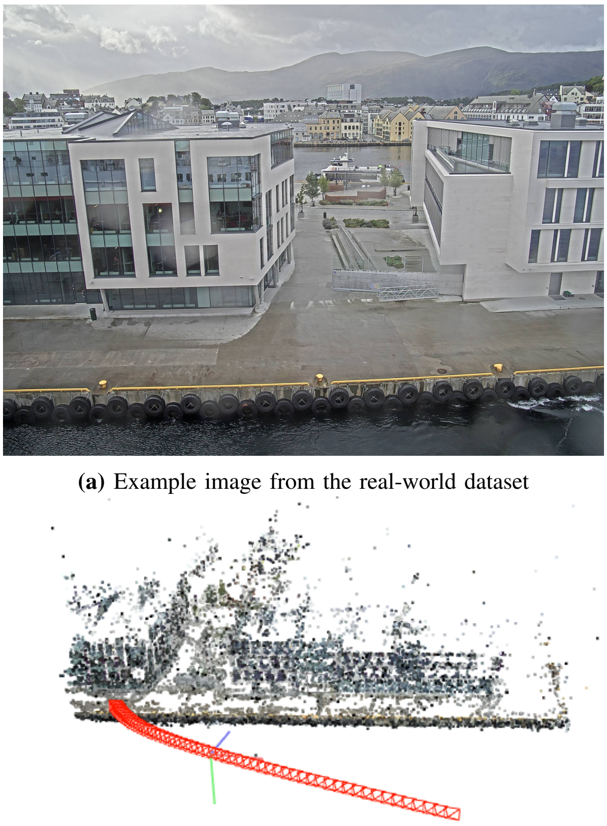


Fig. 2: Randomly generated planar ship-movement



(a) Example image from the real-world dataset

(b) Structure-from-Motion reconstruction of the real-world dataset camera poses

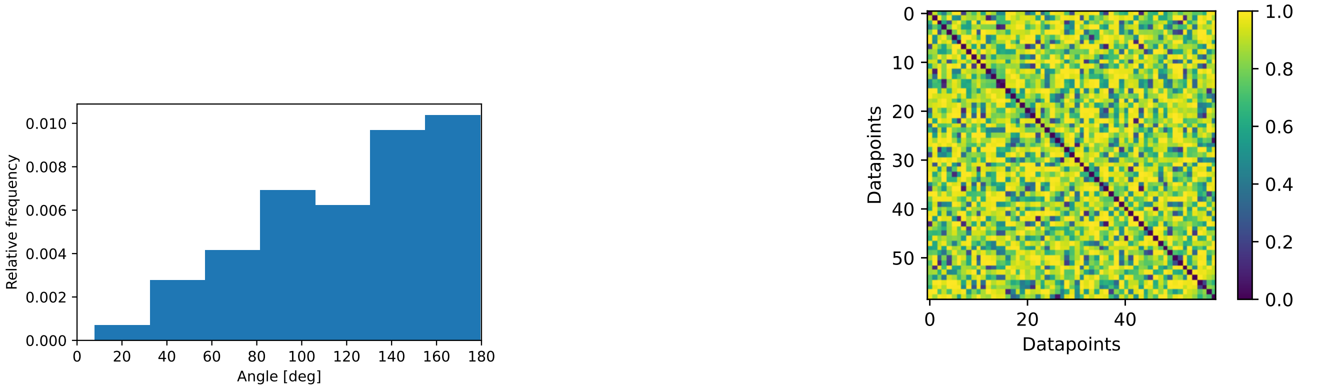
Fig. 3: Illustrations of the real-world dataset used in this work.

uniform dataset shows to have higher excitation of type 1 than the synthetic planar, an intuitive result given the irratic random nature of the dataset. Miles showed in [16] that the rotation angle, θ, of a uniformly distributed rotation will itself be distributed with density $2\sin^2(\frac{\theta}{2})/\pi = (1-\cos(\theta))/\pi$. Comparing this density to the histogram in Figure 4a, which will be an approximation to the probability density of $||\boldsymbol{\omega}_i||^2 = \theta^2$, the results match with the expected outcome reasonably well.

Figures 4b and 5b show cross-plots of $|\sin[\angle(\boldsymbol{\omega}_{\text{nb}}(t_i), \boldsymbol{\omega}_{\text{nb}}(t_j))]|$ for all pairs of timestamps t_i, t_j over the dataset. It is clear that the synthetic uniform dataset has higher excitation of this kind as well.

From these results, one can conclude that the synthetic uniform dataset is more strongly excited than the synthetic planar dataset, given the definitions of excitation made in this work. It is therefore expected that the former dataset should allow for better estimates, since the two datasets are otherwise identical when it comes to noise and size.

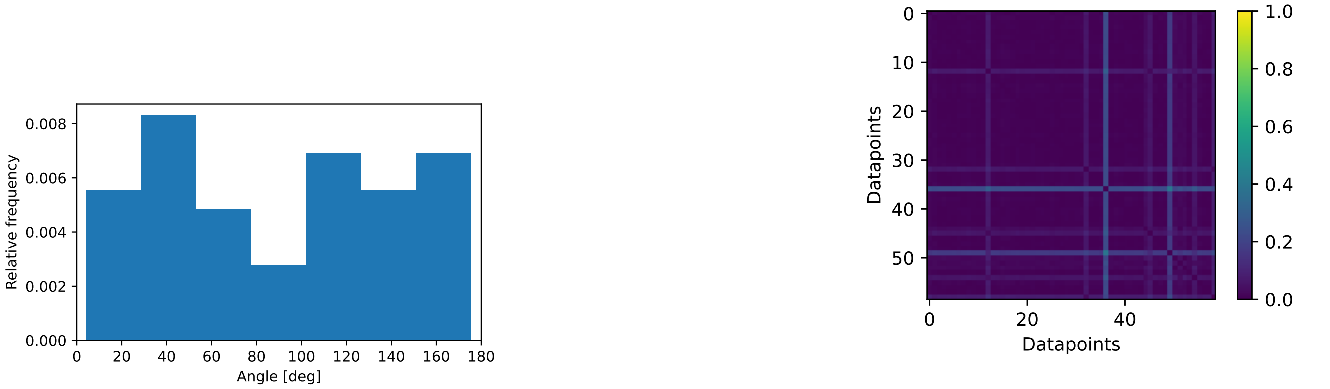
Figures 6a and 6b show the same measure of Hand-Eye excitation, but this time evaluated over the real-world dataset. Given the definitions of excitation presented in this paper, it seems this dataset has comparable type 2 excitation to that of the synthetic planar dataset, but lower type 1 excitation. The low amount of type 1 excitation can be said to reflect the slow and careful movements the ship performed as it was leaving port, see the discussion of Figure 3b in Section V-A. Figure 6a also shows that the cruise ship does not move more than about 15° away from its initial orientation.



(a) Histogram of rotation magnitudes present in the synthetic uniform dataset

(b) Cross-plot of $|\sin[\angle(\boldsymbol{\omega}_{nb}(t_i), \boldsymbol{\omega}_{nb}(t_j))]|$ for all pairs of rotation axes $\boldsymbol{\omega}_{nb}(t_i), \boldsymbol{\omega}_{nb}(t_j)$ in the synthetic uniform dataset

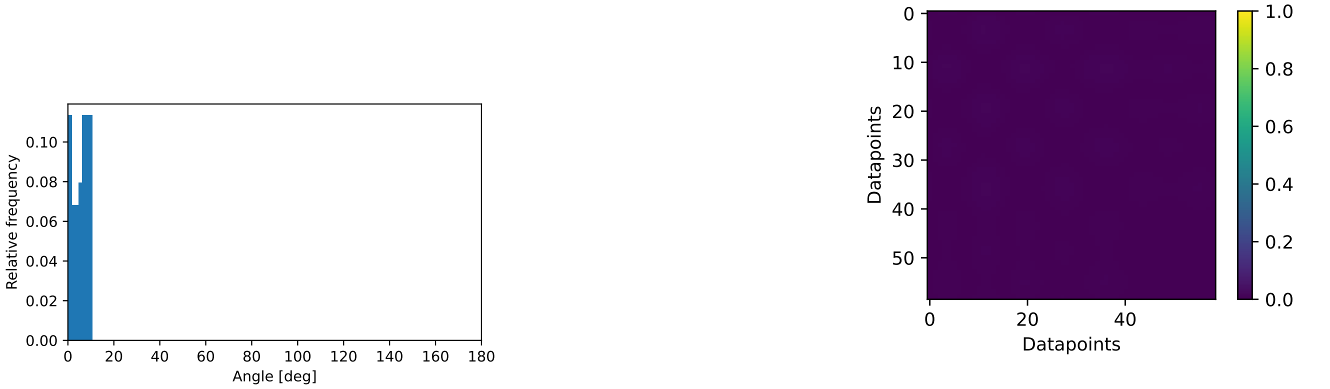
Fig. 4: Visualizing the excitation of the synthetic uniform dataset



(a) Histogram of rotation magnitudes present in the synthetic planar dataset

(b) Cross-plot of $|\sin[\angle(\boldsymbol{\omega}_{nb}(t_i), \boldsymbol{\omega}_{nb}(t_j))]|$ for all pairs of rotation axes $\boldsymbol{\omega}_{nb}(t_i), \boldsymbol{\omega}_{nb}(t_j)$ in the synthetic planar dataset

Fig. 5: Visualizing the excitation of the synthetic planar dataset



(a) Histogram of rotation magnitudes present in the real-world dataset

(b) Cross-plot of $|\sin[\angle(\boldsymbol{\omega}_{nb}(t_i), \boldsymbol{\omega}_{nb}(t_j))]|$ for all pairs of rotation axes $\boldsymbol{\omega}_{nb}(t_i), \boldsymbol{\omega}_{nb}(t_j)$ in the real-world dataset

Fig. 6: Visualizing the excitation of the real-world dataset

C. Performance of the pipeline

Two metrics are used for evaluating the performance of the algorithm pipeline when it comes to its ability to construct accurate estimates. To strengthen the results and avoid “over-fitting”, every even numbered datapoint is fed to the pipeline to generate an estimate, while every odd number is used for testing.

For comparing estimated camera-orientation versus the old extrinsics, a norm over the group of orientations $\mathrm{SO}(3)$ is used as a metric of comparison. The metric returns the angle of the smallest rotation connecting the two orientations, scaled to degrees. To be explicit, the metric is shown in Equation (10). Use of this metric allows for comparing estimated and ground truth extrinsics directly without the challenges posed by attempting to compare Euler-angles.

$$\mathrm{err}_{\mathrm{GT}} = \frac{180}{\pi} ||\mathrm{Log}(\mathbf{R}_{\mathrm{GT}}^{\mathrm{T}} \mathbf{R}_{\mathrm{est}})||_2 \qquad (10)$$

Since the estimation of the extrinsic parameters are based on finding a solution to the equation $\mathbf{AX} = \mathbf{XB}$, a second reasonable metric for comparing estimates is to substitute the estimate into the equation and see to which degree the equality holds for some given data. Since this work focuses on the orientation of cameras, the sides of the equation may be compared also using the presented $\mathrm{SO}(3)$-norm. The error is averaged over all $(\mathbf{A}, \mathbf{B})$-pairs in the dataset and the error is also scaled to degrees for consistency. The precise definition of the metric is seen in Equation (11).

$$\mathrm{err}_{\mathrm{HE}} = \frac{180}{\pi} \frac{1}{N} \sum_{i=1}^{N} ||\mathrm{Log}\left[(\mathbf{R}_{\mathrm{A},i} \mathbf{R}_{\mathrm{X}})^{\mathrm{T}} (\mathbf{R}_{\mathrm{X}} \mathbf{R}_{\mathrm{B},i}) \right] ||_2 \qquad (11)$$

Tables I and II show simulation results of the pipeline on the synthetic uniform and synthetic planar datasets, respectively. The different solvers are evaluated over 100 random initial extrinsic parameter guesses, with both mean and population variance of both metrics calculated with the generated extrinsic parameter estimate. This is the reason why the Park-Martin closed-form solution has variance equal to zero, being unaffected by choice of initial estimate. The results firstly show all solvers to have results multiple orders of magnitude better on the synthetic uniform than the planar dataset. This may be a reflection of the synthetic uniform dataset being seemingly more highly excited, as measured by the presented qualitative measure. It is also clear that the pipeline gives good estimates for noiseless planar data, with estimates within $0.01°$ of the ground-truth values.

Table III shows the same results, but with the real-world dataset as input. The estimated extrinsic parameters are now significantly further away from the assumed ground-truth parameters, with an error of about $2°$. On the other hand, since the precision of the ground-truth extrinsic parameters given with the dataset is unknown, it is not possible to say if this result is because the estimates are closer to the actual physical orientation or not. The fact that all estimates result in lower “Hand-Eye error” than the old ground-truth may suggest the contrary; that the estimates in fact are better reflections of the actual camera-orientation than the old extrinsics. Whether the lower excitation is to blame for the higher variance obtained with this dataset compared to the synthetic planar dataset is not possible to say, since the real-world dataset has unknown amount of noise.

HE-solver	$\mathrm{err}_{\mathrm{GT}}$ [°]		$\mathrm{err}_{\mathrm{HE}}$ [°]	
	Mean	***Variance***	***Mean***	***Variance***
SO3 opt.	7.930E-07	1.726E-13	1.046E-06	2.999E-13
PM opt.	4.856E-07	5.882E-14	6.378E-07	1.015E-13
PM c.-f.	1.930E-13	5.735E-57	2.562E-13	2.549E-57
AHE opt.	3.994E-07	3.929E-14	5.264E-07	6.830E-14

TABLE I: Performance of the pipeline over the synthetic uniform dataset. The results are generated with 100 random initial values for the iterative solvers.

HE-solver	$\mathrm{err}_{\mathrm{GT}}$ [°]		$\mathrm{err}_{\mathrm{HE}}$ [°]	
	Mean	***Variance***	***Mean***	***Variance***
SO3 opt.	1.510E-02	1.215E-04	2.222E-05	2.631E-10
PM opt.	1.054E-02	6.032E-05	1.553E-05	1.309E-10
PM c.-f.	6.892E-07	1.121E-44	4.601E-09	0.000E+00
AHE opt.	8.909E-03	3.866E-05	1.311E-05	8.370E-11

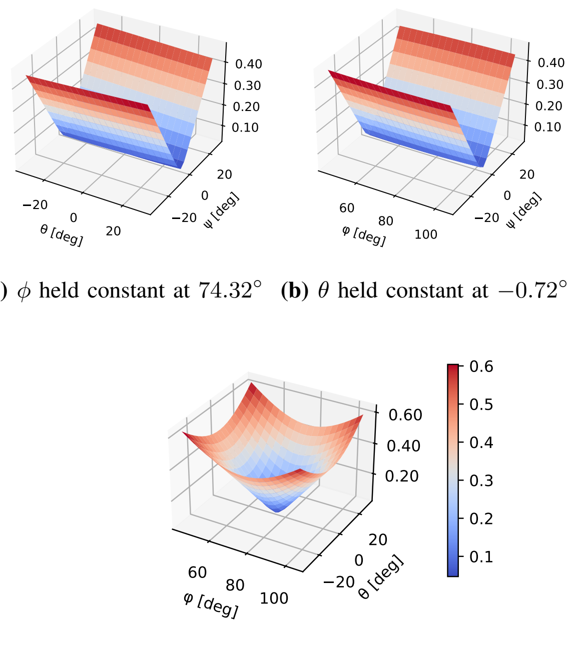
TABLE II: Performance of the pipeline over the synthetic planar dataset. The results are generated with 100 random initial values for the iterative solvers.

The previous results suggest the presented algorithm indeed being able to discern the orientation of a camera with comparable - or better - accuracy than manual methods. It is therefore of interest to analyze the numerical properties of these methods. The Park-Martin residual function has a direct tie to the definition of excitation presented in Section III-D, as well as relatively low variance in Table III. For these reasons, the full cost-function based on the Park-Martin residual was plotted over the real-world dataset, seen in Figure 7. The camera orientation is here described with Euler-angles ϕ, θ, ψ following the zyx convention [17]. For each subplot, one of the optimization variables is kept constant at the old assumed ground truth value, as to enable plotting the cost as the height at a given point in the parameter space. The cost-function over this dataset is seemingly convex and lacks apparent local minima, but in Figures 7a and 7b the cost is almost only

HE-solver	$\mathrm{err}_{\mathrm{GT}}$ [°]		$\mathrm{err}_{\mathrm{HE}}$ [°]	
	Mean	***Variance***	***Mean***	***Variance***
SO3 opt.	1.847E+00	3.509E-02	1.823E-01	8.615E-09
PM opt.	2.010E+00	3.819E-04	1.823E-01	7.912E-12
PM c.-f.	2.009E+00	7.889E-31	1.823E-01	3.081E-33
AHE opt.	2.009E+00	8.805E-05	1.823E-01	1.801E-12
Org. ext.	N/A	N/A	2.463E-01	N/A

TABLE III: Performance of the pipeline over the real-world dataset. The results are generated with 100 random initial values for the iterative solvers. The results from inserting the original (assumed correct) extrinsic parameters are also shown. Note the apparent lower HE-error in the new estimates compared to the old. Note also the very similar mean metric values for the different residuals. These residuals in fact resulted in metrics within 10^{-5} of each other.

dependent on changes in the ψ Euler-angle. Further testing gave similar shape for the synthetic planar dataset, suggesting the planarity common between the two datasets to cause the shape.



(a) ϕ held constant at $74.32°$ **(b)** θ held constant at $-0.72°$

(c) ψ held constant at $-1.77°$

Fig. 7: Sketch of the Park-Martin residual, with the real-world dataset as input

VI. CONLUSION AND FURTHER WORK

Our work tested combining egomotion reconstruction-algorithms of camera-movement with solvers of the Hand-Eye calibration problem, for the purposes of estimating the orientation of ship-mounted cameras.

The performance of the developed algorithm was evaluated on both synthetic and real data. The tests showed that the algorithm was capable of estimating the orientation of cameras within 2° of the old extrinsics, and with better performance than the old extrinsics in a data-driven metric.

A qualitative measure was also presented for the purpose of enabling easier understanding of the amount of excitation in data input to the Hand-Eye calibration methods. Testing the method gave overall expected results, but further work can be done on developing theory on the matter of Hand-Eye excitation, as well as finding ways to employ this in data-selection strategies.

REFERENCES

[1] Y. Shiu and S. Ahmad, “Calibration of wrist-mounted robotic sensors by solving homogeneous transform equations of the form AX=XB,” *IEEE Transactions on Robotics and Automation*, vol. 5, no. 1, pp. 16–29, 1989.

[2] A. Tabb and K. M. A. Yousef, “Solving the robot-world hand-eye(s) calibration problem with iterative methods,” *CoRR*, vol. abs/1907.12425, 2019.

[3] N. Andreff, R. Horaud, and B. Espiau, “Robot hand-eye calibration using structure-from-motion,” *The International Journal of Robotics Research*, vol. 20, no. 3, pp. 228–248, 2001.

[4] N. H. Khan and A. Adnan, “Ego-motion estimation concepts, algorithms and challenges: an overview,” *Multimedia Tools and Applications*, vol. 76, pp. 16581–16603, 2017.

[5] D. Yang, B. He, M. Zhu, and J. Liu, “An extrinsic calibration method with closed-form solution for underwater opti-acoustic imaging system,” *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 9, pp. 6828–6842, 2020.

[6] N. Roy, P. Newman, and S. Srinivasa, “Extrinsic calibration from per-sensor egomotion,” in *Robotics: Science and Systems VIII*, pp. 25–32, The MIT Press, 2013.

[7] H. Longuet-Higgins, “A computer algorithm for reconstructing a scene from two projections,” in *Readings in Computer Vision* (M. A. Fischler and O. Firschein, eds.), pp. 61–62, San Francisco (CA): Morgan Kaufmann, 1987.

[8] R. Tsai and R. Lenz, “A new technique for fully autonomous and efficient 3D robotics hand/eye calibration,” *IEEE Transactions on Robotics and Automation*, vol. 5, no. 3, pp. 345–358, 1989.

[9] A. Mertan, D. J. Duff, and G. Unal, “Single image depth estimation: An overview,” *Digital Signal Processing*, vol. 123, p. 103441, 2022.

[10] J. Solà, J. Deray, and D. Atchuthan, “A micro Lie theory for state estimation in robotics,” *CoRR*, vol. abs/1812.01537, 2018.

[11] F. Park and B. Martin, “Robot sensor calibration: solving AX=XB on the Euclidean group,” *IEEE Transactions on Robotics and Automation*, vol. 10, no. 5, pp. 717–721, 1994.

[12] J. L. Schönberger and J.-M. Frahm, “Structure-from-motion revisited,” in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[13] Mapillary, “Opensfm.” https://opensfm.org/. Accesssed January 2023.

[14] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors, “SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python,” *Nature Methods*, vol. 17, pp. 261–272, 2020.

[15] K. Shoemake, “III.6 - uniform random rotations,” in *Graphics Gems III (IBM Version)* (D. KIRK, ed.), pp. 124–132, San Francisco: Morgan Kaufmann, 1992.

[16] R. E. Miles, “On random rotations in R^3,” *Biometrika*, vol. 52, no. 3/4, pp. 636–639, 1965.

[17] T. I. Fossen, “Kinematics,” in *Handbook of Marine Craft Hydrodynamics and Motion Control*, ch. 2, pp. 15–44, John Wiley & Sons, Ltd, 2011.

NTNU

Norwegian University of
Science and Technology