On learning pseudo-sensors to improve egomotion estimation for mobile autonomy

by

Valentin Peretroukhin

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

Abstract

On learning pseudo-sensors to improve egomotion estimation for mobile autonomy

Valentin Peretroukhin
Doctor of Philosophy
Graduate Department of Institute for Aerospace Studies
University of Toronto
2019

The ability to estimate *egomotion*, that is, to track one's own pose through an unknown environment, is at the heart of safe and reliable mobile autonomy. By inferring pose changes from sequential sensor measurements, egomotion estimation forms the basis of mapping and navigation pipelines, and permits mobile robots to self-localize within environments where external localization sources are intermittent or unavailable. Visual and inertial egomotion estimation, in particular, have become ubiquitous in mobile robotics due to the availability of high-quality, compact, and inexpensive sensors that capture rich representations of the world. To remain computationally tractable, 'classical' visual-inertial pipelines (like visual odometry and visual SLAM) make simplifying assumptions that, while permitting reliable operation in ideal conditions, often lead to systematic error. In this thesis, we present several data-driven learned *pseudo-sensors* that serve to augment conventional pipelines by inferring latent information from the same sensor data. Our approach retains much of the benefits of traditional pipelines, while leveraging high-capacity hyper-parametric models to extract complementary information that can be used to improve uncertainty quantification, correct for systematic bias, and improve robustness to difficult-to-model deleterious effects. We validate our pseudo-sensors on several kilometres of sensor data collected in sundry settings such as urban roads, indoor labs, and planetary analogue sites in the Canadian High Arctic.

Epigraph

A little learning is a dangerous thing;
drink deep, or taste not the Pierian
spring: there shallow draughts
intoxicate the brain, and drinking
largely sobers us again.

Alexander Pope

The universe is no narrow thing and the order within it is not constrained by any latitude in its conception to repeat what exists in one part in any other part. Even in this world more things exist without our knowledge than with it and the order in creation which you see is that which you have put there, like a string in a maze, so that you shall not lose your way. For existence has its own order and that no man's mind can compass, that mind itself being but a fact among others.

Cormac McCarthy

Elephants don't play chess.

Rodney Brooks

To all those who encouraged (or, at least, *never discouraged*) my intellectual wanderlust.

Acknowledgements

This document would not have been possible without the generous support and guidance of my supervisor[1], the perennial love of my family and friends[2], and the limitless patience of my lab mates[3]. Thank you all.

[1] as well as all of my collaborators and academic mentors (special thanks to Lee)
[2] especially the support and encouragement of Elyse
[3] in humouring my insatiable need for debate and banter (special thanks to Lee)

Contents

Notation

a	:	Symbols in this font are real scalars.
$\mathbf{a}$	:	Symbols in this font are real column vectors.
$\mathbf{A}$	:	Symbols in this font are real matrices.
$\mathcal{N}(\boldsymbol{\mu}, \mathbf{R})$	:	Normally distributed with mean $\boldsymbol{\mu}$ and covariance $\mathbf{R}$.
$E[\cdot]$	:	The expectation operator.
$\underrightarrow{\boldsymbol{\mathcal{F}}}_a$	:	A reference frame in three dimensions.
$(\cdot)^\wedge$	:	An operator associated with the Lie algebra for rotations and poses. It produces a matrix from a column vector.
$(\cdot)^\vee$	:	The inverse operation of $(\cdot)^\wedge$
$\mathbf{1}$	:	The identity matrix.
$\mathbf{0}$	:	The zero matrix.
$\mathbf{p}_a^{c,b}$	:	A vector from point b to point c (denoted by the superscript) and expressed in $\underrightarrow{\boldsymbol{\mathcal{F}}}_a$ (denoted by the subscript).
$\mathbf{C}_{a,b}$	:	The 3×3 rotation matrix that transforms vectors from $\underrightarrow{\boldsymbol{\mathcal{F}}}_b$ to $\underrightarrow{\boldsymbol{\mathcal{F}}}_a$: $\mathbf{p}_a^{c,b} = \mathbf{C}_{a,b}\mathbf{p}_b^{c,b}$.
$\mathbf{T}_{a,b}$	:	The 4×4 transformation matrix that transforms homogeneous points from $\underrightarrow{\boldsymbol{\mathcal{F}}}_b$ to $\underrightarrow{\boldsymbol{\mathcal{F}}}_a$: $\mathbf{p}_a^{c,a} = \mathbf{T}_{a,b}\mathbf{p}_b^{c,b}$.

Chapter 4

Predictive Robust Estimation

> Information is the resolution of uncertainty.
>
> Claude Shannon

4.1 Introduction

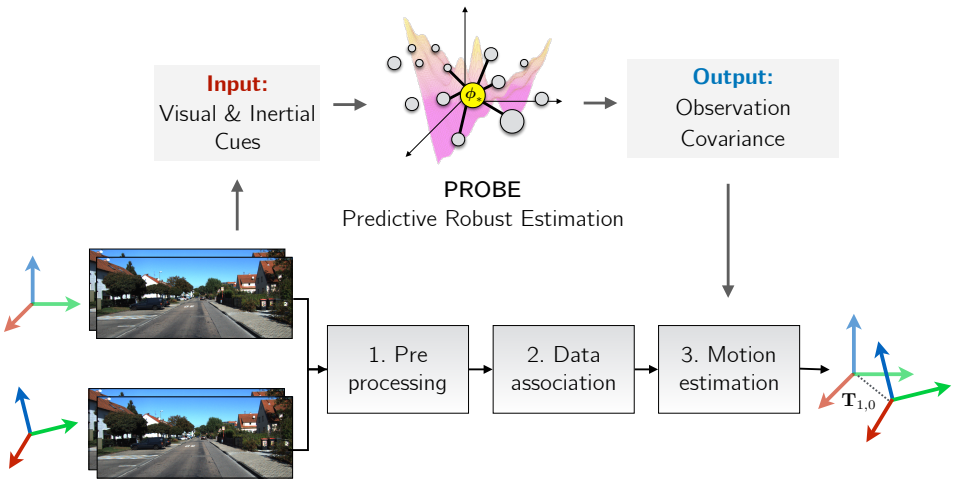

Figure 4.1: PROBE builds a predictive noise model for stereo visual odometry.

The first pseudo-sensor we present is a technique we call PRedictive ROBust Estimation, or PROBE. This approach uses non-parametric learning to build a model for anisotropic observation covariances for a stereo visual odometry pipeline. Namely, we apply the method of Generalized Kernels to a Bayesian treatment of covariance estimation. We show that by assuming a particular covariance prior over re-projection errors, we can then naturally derive

a robust least squares objective that resembles the widely-used Cauchy loss. The parameters of this robust loss are predicted (hence *predictive* robust estimation) for each error term as a function of a prediction space that we define.

PROBE was initially published as a simpler non-Bayesian technique that learned isotropic covariances through a k-nearest-neighbours approach (see Appendix A for more details). The following two publications summarize this initial technique:

1. Peretroukhin, V., Clement, L., Giamou, M., and Kelly, J. (2015a). PROBE: Predictive robust estimation for visual-inertial navigation. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'15)*, pages 3668–3675, Hamburg, Germany

2. Peretroukhin, V., Clement, L., and Kelly, J. (2015b). Get to the point: Active covariance scaling for feature tracking through motion blur. In *Proceedings of the IEEE International Conference on Robotics and Automation Workshop on Scaling Up Active Perception*, Seattle, Washington, USA

We significantly extended this technique to full anisotropic covariances and generalized kernels in the following publication:

1. Peretroukhin, V., Vega-Brown, W., Roy, N., and Kelly, J. (2016). PROBE-GK: Predictive robust estimation using generalized kernels. In *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, pages 817–824 .

We will present this latter technique in this chapter.

4.2 Motivation

Robot navigation relies on an accurate quantification of sensor noise or uncertainty in order to produce reliable state estimates. In practice, this uncertainty is often fixed for a given sensor and experiment, whether by automatic calibration or by manual tuning. Although a fixed measure of uncertainty may be reasonable in certain static environments, dynamic scenes frequently exhibit many effects that corrupt a portion of the available observations. For visual sensors, these effects include, for example, self-similar textures, variations in lighting, moving objects, and motion blur. Further, there may be useful information available in these observations that would normally be rejected by a fixed-threshold outlier rejection scheme. Ideally, we would like to retain some of these observations in our estimator, while still placing more trust in observations that do not suffer from such effects.

4.3 Related Work

There is a large and growing body of work on the problem of deriving accurate, consistent state estimates from visual data. Although our approach to noise modelling is applicable in other domains, for simplicity we focus our attention on the problem of inferring egomotion from features extracted from sequential pairs of stereo images.

Apart from simply rejecting outliers, a number of recent approaches attempt to select the optimal set of features to produce an accurate localization estimate from tracked visual features. For example, Tsotsos et al. (2015) amend Random Sample Consensus (RANSAC) with statistical hypothesis testing to ensure that tracked visual features have normally distributed residuals before including them in the estimator. Unlike our predictive approach, their technique relies on the availability of feature tracks, and requires scene overlap to work continuously. In a different approach, Zhang and Vela (2015) choose an optimally observable feature subset for a monocular SLAM pipeline by selecting features with the highest *informativeness* - a measure calculated based on the observability of the SLAM subsystem. Observability, however, is governed by the 3D location of the features, and therefore cannot predict systematic feature degradation due to environmental or sensor-based effects.

4.4 Predictive Robust Estimation for VO

We present a principled, data-driven way to build a noise model for visual odometry. We leverage recent advances in covariance estimation (Vega-Brown and Roy, 2013) to formulate a predictive robust estimator for a stereo visual odometry pipeline. We frame the traditional non-linear least squares optimization problem as a problem of maximum likelihood estimation with a Gaussian noise model, and infer a distribution over the covariance matrix of the Gaussian noise from a predictive model learned from training data. This results in a Student's t distribution over the noise, and naturally yields a robust nonlinear least-squares optimization problem. In this way, we can predict, in a principled manner, how informative each visual feature is with respect to the final state estimate, which allows our approach to intelligently weight observations to produce more accurate odometry estimates.

4.4.1 Bayesian Noise Model for Visual Odometry

We adopt the motion solution for visual odometry based on reprojection errors presented in Section 3.2.3. In brief, this technique assumes independent Gaussian errors on stereo repro-

jections of a landmark from one frame, $\underrightarrow{\mathcal{F}}_{c_0}$ into a subsequent frame, $\underrightarrow{\mathcal{F}}_{c_1}$:

$$\mathbf{e}_i(\mathbf{T}_t) = \mathbf{e}_{i,t} = \mathbf{y}_{i,c_1} - \mathbf{f}(\mathbf{T}_t \mathbf{f}^{-1}(\mathbf{y}_{i,c_0})) \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_{i,t}). \tag{4.1}$$

Maximizing the likelihood of these errors is then equivalent to solving the following weighted non-linear least squares objective for $\mathbf{T}_t \in \mathrm{SE}(3)$ the rigid-body transform that transforms points in $\underrightarrow{\mathcal{F}}_{c_0}$ to those in $\underrightarrow{\mathcal{F}}_{c_1}$:

$$\mathbf{T}_t^* = \underset{\mathbf{T} \in \mathrm{SE}(3)}{\operatorname{argmax}} \prod_{i=1}^{N_t} p(\mathbf{e}_{i,t}; \mathbf{T}_t, \mathbf{R}_{i,t}) \tag{4.2}$$

$$= \underset{\mathbf{T} \in \mathrm{SE}(3)}{\operatorname{argmin}} \sum_{i=1}^{N_t} {\mathbf{e}_{i,t}}^T \mathbf{R}_{i,t}^{-1} \mathbf{e}_{i,t}. \tag{4.3}$$

With PROBE, instead of treating $\mathbf{R}_{i,t}$ as fixed, we build a model for it as a function of some useful *predictor*, $\boldsymbol{\phi}_{i,t}$,

$$\mathbf{R}_{i,t} = \mathbf{R}(\boldsymbol{\phi}_{i,t}). \tag{4.4}$$

Each predictor can be computed based on the stereo track ($\{\mathbf{y}_{i,c_0}, \mathbf{y}_{i,c_1}\}$) and additional visual[1] and inertial cues, allowing us to model effects like motion blur and self-similar textures. Further, instead of treating the covariance as a point function $\mathbf{R}(\boldsymbol{\phi}_{i,t})$, we instead build a non-parametric model of covariance *density* based on a training dataset, $\mathcal{D}$,

$$p(\mathbf{R}_{i,t}) = p(\mathbf{R}|\mathcal{D}, \boldsymbol{\phi}_{i,t}). \tag{4.5}$$

We will seek the transform that then maximizes the posterior predictive distribution of the errors, given this posterior:

$$\mathbf{T}_t^* = \underset{\mathbf{T} \in \mathrm{SE}(3)}{\operatorname{argmax}} \prod_{i=1}^{N_t} \int p(\mathbf{e}_{i,t}; \mathbf{T}_t|\mathbf{R}_{i,t}) p(\mathbf{R}|\mathcal{D}, \boldsymbol{\phi}_{i,t}) d\mathbf{R}. \tag{4.6}$$

Although at first this may seem unwieldy, we present an efficient method for computing the posterior and show that a particular formulation allows it to be marginalized out analytically to arrive at a simple posterior predictive distribution with a straight-forward objective for achieving a maximum likelihood egomotion transform.

[1] Including potentially data from all four images in the pair of stereo images.

4.4.2 Generalized Kernels

To do this, we leverage Generalized Kernel (GK) estimation (Vega-Brown et al., 2014). GK estimation combines the benefits of kernel density estimation with Bayesian inference. The basic idea is as follows. Consider a dataset of inputs, $\mathbf{x}$, and outputs, $\mathbf{y}$, and a dataset of independent observations $\mathcal{D} = \{(\mathbf{x}_1, \mathbf{y}_1), ..., (\mathbf{x}_N, \mathbf{y}_N)\}$. We are given a new 'test' input $\mathbf{x}^*$, and are asked to infer the likelihood of a observing a given output at this input:

$$p(\mathbf{y}|\mathbf{x}^*, \mathcal{D}). \tag{4.7}$$

If we associate a set of latent parameters, $\boldsymbol{\pi}$, with each input $\mathbf{x}$, and assume a known likelihood function $p(\mathbf{y}|\boldsymbol{\pi})$, we can infer a distribution over $\boldsymbol{\pi}$ and then marginalize it out to arrive at the desired likelihood

$$p(\mathbf{y}|\mathbf{x}^*, \mathcal{D}) = \int_{\boldsymbol{\pi}} \underbrace{p(\mathbf{y}|\boldsymbol{\pi}^*)}_{\text{Known likelihood function}} \underbrace{p(\boldsymbol{\pi}^*|\mathbf{x}^*, \mathcal{D})}_{\text{Parameter posterior}} d\boldsymbol{\pi}^*. \tag{4.8}$$

This is called the posterior predictive distribution. Using Bayes rule, we can write

$$p(\boldsymbol{\pi}^*|\mathbf{x}^*, \mathcal{D}) \propto \int \left(\prod_{i=1}^{N} p(\mathbf{y}_i|\boldsymbol{\pi}_i) d\boldsymbol{\pi}_i \right) p(\boldsymbol{\pi}_{1:N}, \boldsymbol{\pi}^*|\mathbf{x}^*, \mathbf{x}_{1:N}) \tag{4.9}$$

The technique of generalized kernels makes the assumption that the parameters $\boldsymbol{\pi}_{1:N}$ are conditionally independent given the target parameters, $\boldsymbol{\pi}^*$. This gives the distribution:

$$p(\boldsymbol{\pi}_{1:N}, \boldsymbol{\pi}^*|\mathbf{x}^*, \mathbf{x}_{1:N}) = \left(\prod_{i=1}^{N} p(\boldsymbol{\pi}_i|\boldsymbol{\pi}^* \mathbf{x}^*, \mathbf{x}_i) \right) p(\boldsymbol{\pi}^*|\mathbf{x}^*), \tag{4.10}$$

which combined with Equation (4.9) results in

$$p(\boldsymbol{\pi}^*|\mathbf{x}^*, \mathcal{D}) \propto \prod_{i=1}^{N} \underbrace{p(\mathbf{y}_i|\boldsymbol{\pi}^*, \mathbf{x}_i, \mathbf{x}^*)}_{\text{Extended likelihood}} \underbrace{p(\boldsymbol{\pi}^*|\mathbf{x}^*)}_{\text{Prior}}. \tag{4.11}$$

Now, the pièce de résistance of generalized kernels is that the *extended* likelihood can be written as function of the known likelihood $p(\mathbf{y}_i|\boldsymbol{\pi}_i)$ if we assume it is the maximum entropy distribution whose information divergence from the likelihood is bounded by the metric $\rho(\mathbf{x}^*, \mathbf{x}_i)$. Specifically, in Vega-Brown et al. (2014), it is shown that under these assumptions, the extended likelihood must have the form:

$$p(\mathbf{y}|\boldsymbol{\pi}^*, \mathbf{x}, \mathbf{x}^*) \propto p(\mathbf{y}|\boldsymbol{\pi})^{k(\mathbf{x}^*, \mathbf{x})}, \tag{4.12}$$

where $k(\cdot,\cdot)$ is a kernel function[2] that is uniquely defined by ρ. The intuition behind this is that we expect the extended likelihood to equal the known likelihood if $\mathbf{x}^* = \mathbf{x}_i$ (and therefore $\boldsymbol{\pi}^* = \boldsymbol{\pi}_i$, resulting in $p(\mathbf{y}_i|\boldsymbol{\pi}^*, \mathbf{x}_i, \mathbf{x}^*) = p(\mathbf{y}_i|\boldsymbol{\pi}_i)$) and diverge in some smooth way when $\mathbf{x}^* \neq \mathbf{x}_i$. Combining Equation (4.11) with Equation (4.12), we arrive at an expression for the posterior over parameters as

$$p(\boldsymbol{\pi}|\mathbf{x}, \mathcal{D}) \propto \prod_{i=1}^{N} p(\mathbf{y}|\boldsymbol{\pi})^{k(\mathbf{x},\mathbf{x}_i)} p(\boldsymbol{\pi}|\mathbf{x}), \tag{4.13}$$

which can be evaluated in closed form for appropriate an appropriate likelihood and prior. Namely, for PROBE, we will assume Gaussian likelihoods for the reprojection errors (and therefore the observations $\mathbf{y}_{i,c_1}$), and inverse Wishart priors for covariance matrices (this will result in inverse Wishart posteriors due to conjugacy). The input, $\mathbf{x}$, will be the vector of predictors $\boldsymbol{\phi}$.

4.4.3 Generalized Kernels for Visual Odometry

In order to exploit conjugacy to a Gaussian noise model, we formulate our prior knowledge about this function using an inverse Wishart (IW) distribution over positive definite $d \times d$ matrices (the IW distribution has been used as a prior on covariance matrices in other robotics and computer vision contexts, see for example, (Fitzgibbon et al., 2007)). This distribution is defined by a scale matrix $\boldsymbol{\Psi} \in \mathbb{R}^{d\times d} \succ 0$ and a scalar quantity called the degrees of freedom $\nu \in \mathbb{R} > d-1$:

$$\begin{aligned} p(\mathbf{R}) &= \mathrm{IW}(\mathbf{R}; \boldsymbol{\Psi}, \nu) \\ &= \frac{|\boldsymbol{\Psi}|^{\nu/2}}{2^{\frac{\nu d}{2}} \Gamma_d(\frac{\nu}{2})} |\mathbf{R}|^{-\frac{\nu+d+1}{2}} \exp\left(-\frac{1}{2}\mathrm{tr}\left(\boldsymbol{\Psi}\mathbf{R}^{-1}\right)\right). \end{aligned} \tag{4.14}$$

We use the scale matrix to encode our prior estimate of the covariance, and the degrees of freedom to encode our confidence in that estimate. Specifically, if we estimate the covariance $\mathbf{R}$ associated with predictor $\boldsymbol{\phi}$ to be $\hat{\mathbf{R}}$ with a confidence equivalent to seeing n independent samples of the error from $\mathcal{N}(\mathbf{0}, \hat{\mathbf{R}})$, we would choose $\nu(\boldsymbol{\phi}) = n$ and $\boldsymbol{\Psi}(\boldsymbol{\phi}) = n\hat{\mathbf{R}}$. Given a sequence of observations and ground truth transformations,

$$\mathcal{D} = \{\mathcal{I}_t, \mathbf{T}_t\}, \quad t \in [1, N] \tag{4.15}$$

[2] i.e., $k(\mathbf{x}, \mathbf{x}) = 1 \forall \mathbf{x}$ and $k(\mathbf{x}, \mathbf{x}') \in [0, 1] \forall \mathbf{x}, \mathbf{x}'$.

where

$$\mathcal{I}_t = \{\mathbf{y}_{i,c_0}, \mathbf{y}_{i,c_1}, \boldsymbol{\phi}_{i,t}\} \quad i \in [1, N_t], \tag{4.16}$$

we can use the procedure of generalized kernel estimation as described above to infer a posterior distribution over the covariance matrix $\mathbf{R}_*$ associated with some query predictor vector $\boldsymbol{\phi}_*$:

$$\begin{aligned} p(\mathbf{R}_*|\mathcal{D}, \boldsymbol{\phi}_*) \propto \prod_{i,t} \mathcal{N}(\mathbf{e}_{i,t}|\mathbf{0}, \mathbf{R}_*)^{k(\boldsymbol{\phi}_*, \boldsymbol{\phi}_{i,t})} \\ \times \mathrm{IW}(\mathbf{R}_*; \boldsymbol{\Psi}(\boldsymbol{\phi}_*), \nu(\boldsymbol{\phi}_*)) \qquad (4.17) \\ = \mathrm{IW}(\mathbf{R}_*; \boldsymbol{\Psi}_*, \nu_*). \qquad (4.18) \end{aligned}$$

Here, $\mathbf{e}_{i,t} = \mathbf{y}_{i,c_1} - \mathbf{f}(\mathbf{T}_t \mathbf{f}^{-1}(\mathbf{y}_{i,c_0}))$ as before. The function $k : \mathbb{R}^M \times \mathbb{R}^M \to [0, 1]$ is a kernel function which measures the similarity of two points in predictor space. Note also that the posterior parameters $\boldsymbol{\Psi}_*$ and ν_* can be computed in closed form (see Vega-Brown et al. (2014)) as

$$\boldsymbol{\Psi}_* = \boldsymbol{\Psi}(\boldsymbol{\phi}_*) + \sum_{i,t} k(\boldsymbol{\phi}_*, \boldsymbol{\phi}_{i,t}) \mathbf{e}_{i,t}\, \mathbf{e}_{i,t}{}^T, \tag{4.19}$$

$$\nu_* = \nu(\boldsymbol{\phi}_*) + \sum_{i,t} k(\boldsymbol{\phi}_*, \boldsymbol{\phi}_{i,t}). \tag{4.20}$$

If we marginalize over the covariance matrix, we find that the posterior predictive distribution is a multivariate Student's t distribution:

$$\begin{aligned} p(\mathbf{y}_{i,c_1}|\mathbf{T}_t, \mathbf{y}_{i,c_0}, \mathcal{D}, \boldsymbol{\phi}_{i,t}) \qquad (4.21) \\ = \int \mathrm{d}\mathbf{R}_{i,t} \mathcal{N}\left(\mathbf{e}_{i,t}; \mathbf{0}, \mathbf{R}_{i,t}\right) \mathrm{IW}(\mathbf{R}_{i,t}; \boldsymbol{\Psi}_*, \nu_*) \qquad (4.22) \\ = \mathbf{t}_{\nu_* - d + 1}\left(\mathbf{e}_{i,t}; \mathbf{0}, \frac{1}{\nu_* - d + 1} \boldsymbol{\Psi}_*\right) \qquad (4.23) \\ = \frac{\Gamma(\frac{\nu_*+1}{2})}{\Gamma(\frac{\nu_*-d+1}{2})} |\boldsymbol{\Psi}_*|^{-\frac{1}{2}} \pi^{-\frac{d}{2}} \left(1 + \mathbf{e}_{i,t}{}^T \boldsymbol{\Psi}_*{}^{-1} \mathbf{e}_{i,t}\right)^{-\frac{\nu_*+1}{2}}. \qquad (4.24) \end{aligned}$$

Given a new landmark and predictor vector, we can infer a noise model by evaluating eqs. (4.19) and (4.20). In order to accelerate this computation, it is helpful to choose a kernel function with finite support: that is, $k(\boldsymbol{\phi}, \boldsymbol{\phi}') = 0$ if $\|\boldsymbol{\phi} - \boldsymbol{\phi}'\|_2 > \gamma$ for some γ. Then, by indexing our training data in a spatial index such as a k-d tree, we can identify the subset of samples relevant to evaluating the sums in eqs. (4.19) and (4.20) in $\mathcal{O}(\log N + \log N_t)$ time. Algorithm 1 describes the procedure for building this model.

Algorithm 1 Build the covariance model given a sequence of observations, $\mathcal{D}$.

function BUILDCOVARIANCEMODEL($\mathcal{D}$)
  Initialize an empty spatial index $\mathcal{M}$
  for all $\mathcal{I}_t, \mathbf{T}_t$ in $\mathcal{D}$ **do**
    for all $\{\mathbf{y}_{i,c_0}, \mathbf{y}_{i,c_1}, \boldsymbol{\phi}_{i,c_0}\}$ in $\mathcal{I}_t$ **do**
      $\mathbf{e}_{i,t} = \mathbf{y}_{i,c_1} - \mathbf{f}(\mathbf{T}_t \mathbf{f}^{-1}(\mathbf{y}_{i,c_0}))$
      Insert $\boldsymbol{\phi}_{i,t}$ into $\mathcal{M}$ and store $\mathbf{e}_{i,t}$ at its location
    end for
  end for
  return $\mathcal{M}$
end function

Once we have inferred a noise model for each landmark in a new image pair, the maximum likelihood optimization problem is given by

$$\mathbf{T}_t^* = \underset{\mathbf{T}_t \in \mathrm{SE}(3)}{\operatorname{argmin}} \sum_{i=1}^{N_t} (\nu_{i,t} + 1) \log\left(1 + {\mathbf{e}_{i,t}}^T \boldsymbol{\Psi}_{i,t}^{-1} \mathbf{e}_{i,t}\right). \tag{4.25}$$

The final optimization problem thus emerges as a nonlinear least squares problem with a rescaled Cauchy-like loss function, with error term ${\mathbf{e}_{i,t}}^T (\frac{1}{\nu_{i,t}+1} \boldsymbol{\Psi}_{i,t})^{-1} \mathbf{e}_{i,t}$ and outlier scale $\nu_{i,t} + 1$. This is a common robust loss function which is approximately quadratic in the reprojection error for ${\mathbf{e}_{i,t}}^T \boldsymbol{\Psi}_{i,t}^{-1} \mathbf{e}_{i,t} \ll \nu_{i,t} + 1$, but grows only logarithmically for ${\mathbf{e}_{i,t}}^T \boldsymbol{\Psi}_{i,t}^{-1} \mathbf{e}_{i,t} \gg \nu_{i,t} + 1$. It follows that in the limit of large $\nu_{i,t}$—in regions of predictor space where there are many relevant samples—our optimization problem becomes the original least-squares optimization problem.

Solving nonlinear optimization problems with the form of Equation (4.25) is a well-studied and well-understood task, and software packages to perform this computation are readily available. Algorithm 2 describes the procedure for computing the transform between a new image pair, treating the optimization of Equation (4.25) as a subroutine.

We observe that Algorithm 2 is predictively robust, in the sense that it uses past experiences not just to predict the reliability of a given image landmark, but also to introspect and estimate its own knowledge of that reliability. Landmarks which are not known to be reliable are trusted less than landmarks which look like those which have been observed previously, where "looks like" is defined by our prediction space and choice of kernel.

4.4.4 Inference without ground truth

Algorithm 1 requires access to the true transform between training image pairs. In practice, such ground truth data may be difficult to obtain. In these cases, we can instead formulate a

Algorithm 2 Compute the transform between two images, given a set, $\mathcal{I}_t$, of landmarks and predictors extracted from an image pair and a covariance model $\mathcal{M}$.

function COMPUTETRANSFORM($\mathcal{I}_t$, $\mathcal{M}$)
  for all $\{\mathbf{y}_{i,c_0}, \mathbf{y}_{i,c_1}, \boldsymbol{\phi}_{i,c_0}\}$ in $\mathcal{I}_t$ **do**
    $\boldsymbol{\Psi}, \nu \leftarrow$ INFERNOISEMODEL($\mathcal{M}$, $\boldsymbol{\phi}_{i,t}$)
    $g(\mathbf{T}) = \mathbf{y}_{i,c_1} - \mathbf{f}(\mathbf{T}\mathbf{f}^{-1}(\mathbf{y}_{i,c_0}))$
    $\mathcal{L} \leftarrow \mathcal{L} + (\nu + 1) \log \left(1 + g(\mathbf{T})^T \boldsymbol{\Psi}^{-1} g(\mathbf{T})\right)$
  end for
  return $\operatorname{argmin}_{\mathbf{T} \in \mathrm{SE}(3)} \mathcal{L}(\mathbf{T})$
end function
function INFERNOISEMODEL($\mathcal{M}$, $\boldsymbol{\phi}_*$)
  NEIGHBORS $\leftarrow$ GETNEIGHBORS($\mathcal{M}$, $\boldsymbol{\phi}_*$, ρ) ▷ ρ is the radius of support of kernel k
  $\boldsymbol{\Psi}_* \leftarrow \boldsymbol{\Psi}(\boldsymbol{\phi}_*)$
  $\nu_* \leftarrow \nu(\boldsymbol{\phi}_*)$
  for $(\boldsymbol{\phi}_{i,t}, \mathbf{e}_{i,t})$ in NEIGHBORS **do**
    $\boldsymbol{\Psi}_* \leftarrow \boldsymbol{\Psi}_* + k(\boldsymbol{\phi}_*, \boldsymbol{\phi}_{i,t}) \mathbf{e}_{i,t}\, \mathbf{e}_{i,t}^T$
    $\nu_* \leftarrow \nu_* + k(\boldsymbol{\phi}_*, \boldsymbol{\phi}_{i,t})$
  end for
  return $\boldsymbol{\Psi}_*, \nu_*$
end function

likelihood model $p(\mathcal{D}'|\mathbf{T}_1, \ldots, \mathbf{T}_t)$, where $\mathcal{D}' = \{\mathcal{I}_t\}$ is a dataset consisting only of landmarks and predictors for each training image pair. We can construct a model for future queries by inferring the most likely sequence of transforms for our training images. The likelihood has the following factorized form:

$$p(\mathcal{D}'|\mathbf{T}_{1:T}) \propto \int \prod_{i,t} \mathrm{d}\mathbf{R}_{i,t}\, p(\mathbf{y}_{i,c_1}|\mathbf{y}_{i,c_0}, \mathbf{T}_t, \mathbf{R}_{i,t}) p(\mathbf{R}_{i,t}|\boldsymbol{\phi}_{i,t}, \mathcal{D}, \mathbf{T}_{1:T}). \tag{4.26}$$

We cannot easily maximize this likelihood, since marginalizing over the noise covariances removes the independence of the transforms between each image pair. To render the optimization tractable, we follow previous work (Vega-Brown and Roy, 2013) and formulate an iterative expectation-maximization (EM) procedure. Given an estimate $\mathbf{T}_t^{(n)}$ of the transforms, we can compute the expected log-likelihood conditioned on our current estimate:

$$Q(\mathbf{T}_{1:T}|\mathbf{T}_{1:T}^{(n)}) = \int \left(\prod_{i,t} \mathrm{d}\mathbf{R}_{i,t}\, p(\mathbf{R}_{i,t}|\mathcal{D}_{\setminus i,t}, \mathbf{T}_{1:T}^{(n)}) \right) \log \prod_{i,t} p(\mathbf{y}_{i,c_1}|\mathbf{y}_{i,c_0}, \mathbf{T}_t, \mathbf{R}_{i,t}). \tag{4.27}$$

This has the effect of rendering the likelihood of each transform to be estimated independently. Moreover, the expected log-likelihood can be evaluated in closed form:

$$Q(\mathbf{T}_{1:T}|\mathbf{T}_{1:T}^{(n)}) \cong -\frac{1}{2}\sum_{t=1}^{T}\sum_{i=1}^{N_t} \mathbf{e}_{i,t}{}^T \left(\frac{1}{\nu_{i,t}^{(n)}}\mathbf{\Psi}_{i,t}^{(n)}\right)^{-1} \mathbf{e}_{i,t}. \tag{4.28}$$

The symbol $\cong$ is used to indicate equality up to an additive constant. We can iteratively refine our estimate by maximizing the expected log-likelihood

$$\mathbf{T}_{1:T}^{(n+1)} = \underset{\mathbf{T}_{1:T}\in \mathrm{SE}(3)^T}{\operatorname{argmax}} Q(\mathbf{T}_{1:T}|\mathbf{T}_{1:T}^{(n)}). \tag{4.29}$$

Due to the additive structure of $Q(\mathbf{T}_{1:T}|\mathbf{T}_{1:T}^{(n)})$, this takes the form of T separate nonlinear least-squares optimizations:

$$\mathbf{T}_{t}^{(n+1)} = \underset{\mathbf{T}_t\in \mathrm{SE}(3)}{\operatorname{argmin}} \sum_{i=1}^{N_t} \mathbf{e}_{i,t}{}^T \left(\frac{1}{\nu_{i,t}^{(n)}}\mathbf{\Psi}_{i,t}^{(n)}\right)^{-1} \mathbf{e}_{i,t}. \tag{4.30}$$

Algorithm 3 describes the process of training a model without ground truth. We refer to this process as PROBE-GK-EM, and distinguish it from PROBE-GK-GT (Ground Truth). We note that the sequence of estimated transforms, $\mathbf{T}_{1:T}^{(n)}$, is guaranteed to converge to a local maxima of the likelihood function (Dempster et al., 1977). It is also possible to use a robust loss function (Equation (4.25)) in place of Equation (4.30) during EM training. Although not formally motivated by the derivation above, this approach often leads to lower test errors in practice. Characterizing when and why this robust learning process outperforms its non-robust alternative is outside the scope of this dissertation.

4.5 Prediction Space

A crucial component of our technique is the choice of the vector of predictors $\boldsymbol{\phi}$. In practice, feature tracking quality is often degraded by a variety of effects such as motion blur, moving objects, and textureless or self-similar image regions. The challenge is in determining predictors that account for such effects without requiring excessive computation. In our implementation, we use the following predictors, but stress that the choice of predictors can be tailored to suit particular applications and environments:

- Angular velocity and linear acceleration magnitudes
- Local image entropy

Algorithm 3 Build the covariance model without ground truth given a sequence of observations, $\mathcal{D}'$, and an initial odometry estimate $\mathbf{T}_{1:T}^{(0)}$.

function BUILDCOVARIANCEMODEL($\mathcal{D}'$, $\mathbf{T}_{1:T}^{(0)}$)
  Initialize an empty spatial index $\mathcal{M}$
  for all $\mathcal{I}_t$ in $\mathcal{D}'$ **do**
    for all $\{\mathbf{y}_{i,c_0}, \mathbf{y}_{i,c_1}, \boldsymbol{\phi}_{i,t}\}$ in $\mathcal{I}_t$ **do**
      $\mathbf{e}_{i,t} = \mathbf{y}_{i,c_1} - \mathbf{f}(\mathbf{T}_t^{(0)}\mathbf{f}^{-1}(\mathbf{y}_{i,c_0}))$
      Insert $\boldsymbol{\phi}_{i,t}$ into $\mathcal{M}$ and store $\mathbf{e}_{i,t}$ at its location
    end for
  end for
  repeat
    for all $\mathcal{I}_t$ in $\mathcal{D}'$ **do**
      for all $\{\mathbf{y}_{i,c_0}, \mathbf{y}_{i,c_1}, \boldsymbol{\phi}_{i,t}\}$ in $\mathcal{I}_t$ **do**
        $\boldsymbol{\Psi}, \nu \leftarrow$ INFERNOISEMODEL($\mathcal{M}, \boldsymbol{\phi}_{i,t}$)
        $g(\mathbf{T}) = \mathbf{y}_{i,c_1} - \mathbf{f}(\mathbf{T}\mathbf{f}^{-1}(\mathbf{y}_{i,c_0}))$
        $\mathcal{L} \leftarrow \mathcal{L} + g(\mathbf{T})^T \left(\frac{1}{\nu}\boldsymbol{\Psi}\right)^{-1} g(\mathbf{T})$
      end for
      $\mathbf{T}_t \leftarrow \operatorname{argmin}_{\mathbf{T}\in\mathrm{SE}(3)} \mathcal{L}(\mathbf{T})$
      $\mathbf{e}_{i,t} = \mathbf{y}_{i,c_1} - \mathbf{f}(\mathbf{T}_t^{(0)}\mathbf{f}^{-1}(\mathbf{y}_{i,c_0}))$
      Update the error stored at $\boldsymbol{\phi}_{i,t}$ in $\mathcal{M}$ to $\mathbf{e}_{i,t}$
    end for
  until converged
  return $\mathcal{M}$
end function

- Blur (quantified by the blur metric of Crete et al. (2007))
- Optical flow variance score
- Image frequency composition

We discuss each of these predictors in turn.

4.5.1 Angular velocity and linear acceleration

While most of the predictors in our system are computed directly from image data, the magnitudes of the angular velocities and linear accelerations reported by an IMU (if available) are in themselves good predictors of image degradation (e.g., image blur) and hence poor feature tracking. We do not explicitly correct for bias in linear accelerations because we expect real motion-induced acceleration to trump bias at the timescales of our test trials. As a result, there is virtually no computational cost involved in incorporating these quantities as predictors.

4.5.2 Local image entropy

Entropy is a statistical measure of randomness that can be used to characterize the texture in an image or patch. Since the quality of feature detection is strongly influenced by the strength of the texture in the vicinity of the feature point, we expect the entropy of a patch centered on the feature to be a good predictor of its quality. We evaluate the entropy S in an image patch by sorting pixel intensities into N bins and computing

$$S = -\sum_{i=1}^{N} c_i \log_2(c_i), \tag{4.31}$$

where c_i is the number of pixels counted in the i^{th} bin.

4.5.3 Blur

Blur can arise from a number of sources including motion, dirty lenses, and sensor defects. All of these have deleterious effects on feature tracking quality. To assess the effect of blur in detail, we performed a separate experiment. We recorded images of 32 interior corners of a standard checkerboard calibration target using a low frame-rate (20 FPS) Skybotix VI-Sensor stereo camera and a high frame-rate (125 FPS) Point Grey Flea3 monocular camera rigidly connected by a bar (Figure 4.2). Prior to the experiment, we determined the intrinsic and extrinsic calibration parameters of our rig using the KALIBR[3] package Furgale et al. (2013). The

[3] https://github.com/ethz-asl/kalibr

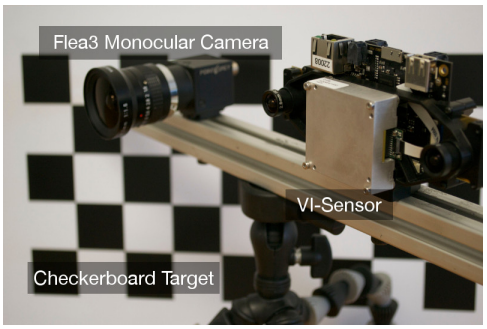

Figure 4.2: The Skybotix VI-Sensor, Point Grey Flea3, and checkerboard target used in our motion blur experiments.

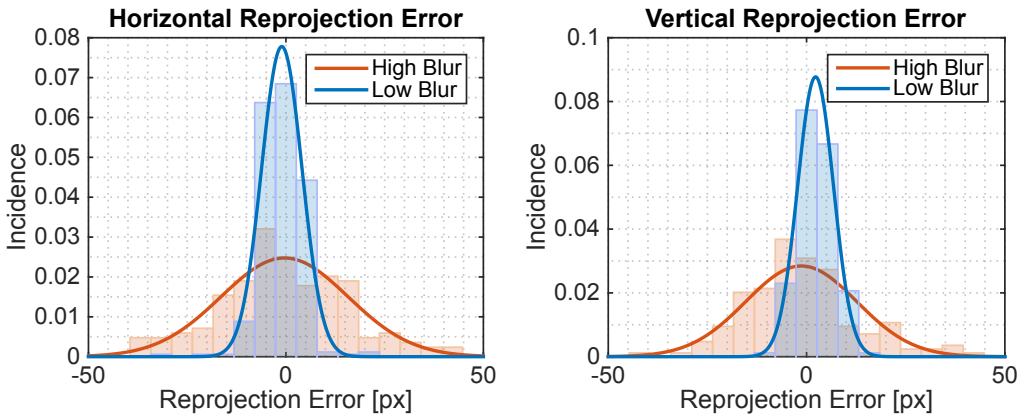

Figure 4.3: Reprojection error of checkerboard corners triangulated from the VI-Sensor and reprojected into the Flea3.We distinguish between high and low blur by thresholding the blur metric Crete et al. (2007).

apparatus underwent both slow and fast translational and rotational motion, which induced different levels of motion blur as quantified by the blur metric proposed by Crete et al. (2007).

We detected checkerboard corners in each camera at synchronized time steps, computed their 3D coordinates in the VI-Sensor frame, then reprojected these 3D coordinates into the Flea3 frame. We then computed the reprojection error as the distance between the reprojected image coordinates and the true image coordinates in the Flea3 frame. Since the Flea3 operated at a much higher frame rate than the VI-Sensor, it was less susceptible to motion blur and so we treated its observations as ground truth. We also computed a tracking error by comparing the image coordinates of checkerboard corners in the left camera of the VI-Sensor computed from both KLT tracking Lucas and Kanade (1981) and re-detection.

Figure 4.4 shows histograms and fitted normal distributions for both reprojection error and tracking error. From these distributions we can see that the errors remain approximately zero-mean, but that their variance increases with blur. This result is compelling evidence that the effect of blur on feature tracking quality can be accounted for by scaling the feature

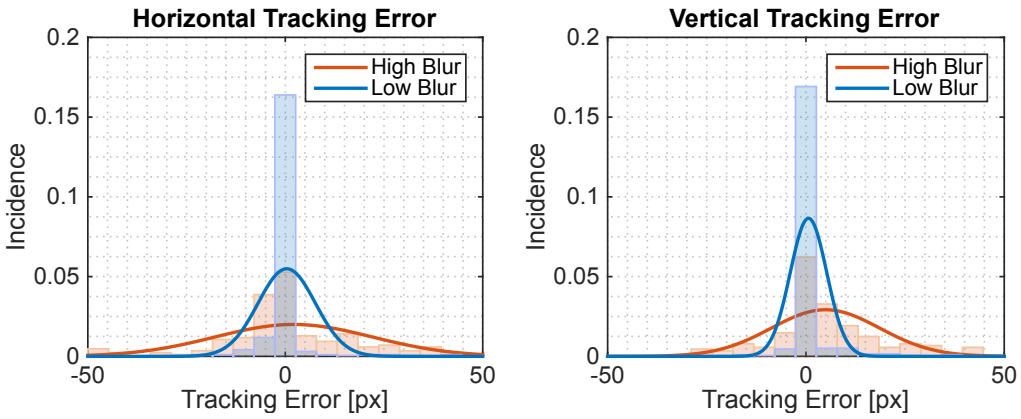

Figure 4.4: Effect of blur on reprojection and tracking error for the slow-then-fast checkerboard dataset. We distinguish between high and low blur by thresholding the blur metric Crete et al. (2007). The variance in both errors increases with blur.

covariance matrix by a function of the blur metric.

4.5.4 Optical flow variance

To detect moving objects, we compute a score for each feature based on the ratio of the variance in optical flow vectors in a small region around the feature to the variance in flow vectors of a larger region. Intuitively, if the flow variance in the small region differs significantly from that in the larger region, we might expect the feature in question to belong to a moving object, and we would therefore like to trust the feature less. Since we consider only the variance in optical flow vectors, we expect this predictor to be reasonably invariant to scene geometry.

We compute this optical flow variance score according to

$$\log\left(\frac{\bar{\sigma}_s^2}{\bar{\sigma}_l^2}\right), \tag{4.32}$$

where $\bar{\sigma}_s^2, \bar{\sigma}_l^2$ are the means of the variance of the vertical and horizontal optical flow vector components in the small and large regions respectively. Figure 4.5 shows sample results of this scoring procedure for two images in the KITTI dataset. Our optical flow variance score generally picks out moving objects such as vehicles and cyclists in diverse scenes.

4.5.5 Image frequency composition

Reliable feature tracking is often difficult in textureless or self-similar environments due to low feature counts and false matches. We detect textureless and self-similar image regions by computing the Fast Fourier Transform (FFT) of each image and analyzing its frequency com-

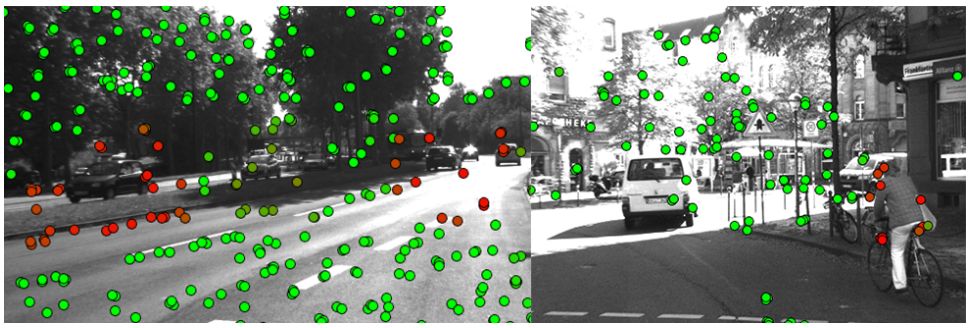

Figure 4.5: The optical flow variance predictor can help in detecting moving objects. Red circles correspond to higher values of the optical flow variance score (i.e., features more likely to belong to a moving object).

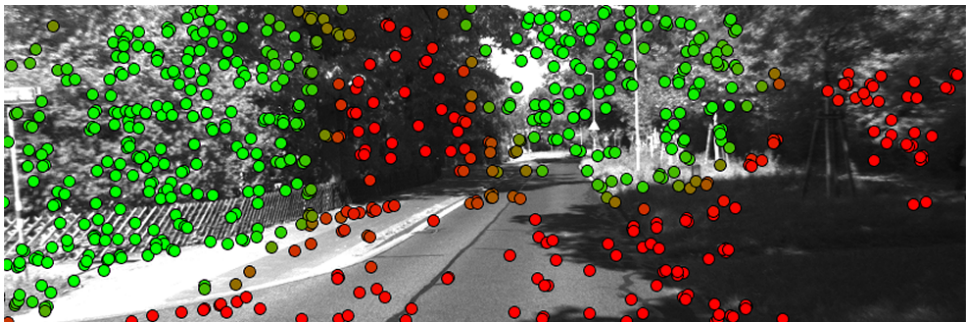

Figure 4.6: A high-frequency predictor can distinguish between regions of high and low texture such as foliage and shadows. Green indicates higher values.

position. For each feature, we compute a coefficient for the low- and high-frequency regimes of the FFT. Figure 4.6 shows the result of the high-frequency version of this predictor on a sample image from the KITTI dataset. Our high-frequency predictor effectively distinguishes between textureless regions (e.g., shadows and roads) and texture-rich regions (e.g., foliage).

4.6 Experiments

To validate PROBE-GK, we used three types of data: synthetic simulations, the KITTI dataset, and our own experimental data collected at the University of Toronto.

4.6.1 Simulation

Monte-Carlo Verification

To begin, we verified that PROBE-GK can predict increasingly accurate estimates of the true error covariance as more training data is added. We developed a basic simulation environment consisting of a large amount of point landmarks being observed by a stereo camera. In

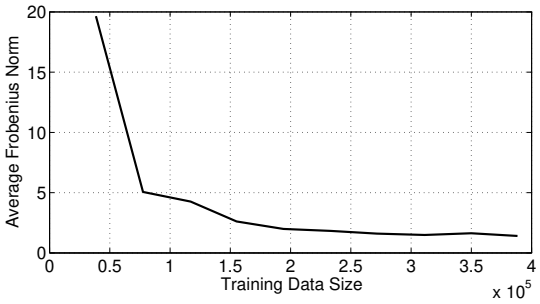

Figure 4.7: Mean Frobenius norm of the error between the estimated and true noise covariance as a function of training data size. The norm tends to zero as training data is added which indicates that PROBE-GK is learning the correct covariances.

our simulation, the camera traversed a single step in one direction, and recorded empirical reprojection errors based on ground truth poses. We simulated additive Gaussian noise on image coordinates, and used Monte Carlo simulations to estimate the true covariances. Figure 4.7 shows the mean Frobenius norm (as defined in Barfoot and Furgale (2014)) between the covariances estimated by PROBE-GK and the true covariances for a test trial. The mean norm tends to zero as more landmarks are added, indicating that PROBE-GK does learn the correct covariances.

Synthetic

Next, we formulated a synthetic dataset wherein a stereo camera traverses a circular path observing 2000 randomly distributed point features. We added Gaussian noise to each of the ideal projected pixel co-ordinates for visible landmarks at every step. We varied the noise variance as a function of the vertical pixel coordinate of the feature in image space. In addition, a small subset of the landmarks received an error term drawn from a uniform distribution to simulate the presence of outliers. The prediction space was composed of the vertical and horizontal pixel locations in each of the stereo cameras.

We simulated independent training and test traversals, where the camera moved for 30 and 60 seconds respectively (at a forward speed of 3 metres per second for final path lengths of 90 and 180 meters). Figure 4.9 and Table 4.1 document the qualitative and quantitative comparisons of PROBE-GK (trained with and without ground-truth) against two baseline stereo odometry frameworks. Both baseline estimators were implemented based on the reprojection-error-based VO pipeline described in Chapter 3. The first utilized fixed covariances for all reprojection errors, while the second used a modified robust cost (i.e. M-estimation) based on Student's t weighting, with $\nu = 5$ (as suggested in Kerl et al. (2013)). These benchmarks served as baseline estimators (with and without robust costs) that used fixed covariance matrices and

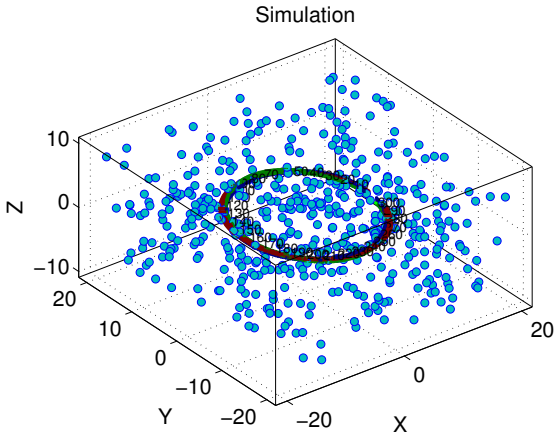

Figure 4.8: Our synthetic world. A stereo camera rig moves through a world with 2000 point features.

did not include a predictive component.

Using PROBE-GK with ground truth data for training, we significantly reduced both the translation and rotational Average Root Mean Squared Error (ARMSE) by approximately 50%. In our synthetic data, the Expectation Maximization approach was able to achieve nearly identical results to the ground-truth-aided model within 5 iterations.

4.6.2 KITTI

To evaluate PROBE-GK on real environments, we trained and tested several models on the KITTI Vision Benchmark suite (Geiger et al., 2013), a series of datasets collected by a car outfitted with a number of sensors driven around different parts of Karlsruhe, Germany. Within the dataset, ground truth pose information is provided by a high grade inertial navigation unit which also fuses measurements from differential GPS. Raw data is available for different types of environments through which the car was driving; for our work, we focused on the city, residential and road categories (Figure 4.10). From each category, we chose two separate trials for training and testing.

Figures 4.11 to 4.13 show typical results; Table 4.1 presents a quantitative comparison. PROBE GK-GT produced significant reductions in ARMSE, reducing translational ARMSE by as much as 80%. In contrast, GK-EM showed more modest improvements; this is unlike our synthetic experiments, where both GK-EM and GK-GT achieved similar performance. We note that although our simulated data is drawn from a mixture of Gaussian distributions, the underlying noise distribution for real data may be far more complex. With no ground truth, EM has to jointly optimize the camera poses and sensor uncertainty. It is unclear whether this is feasible in the general case with no ground truth information.

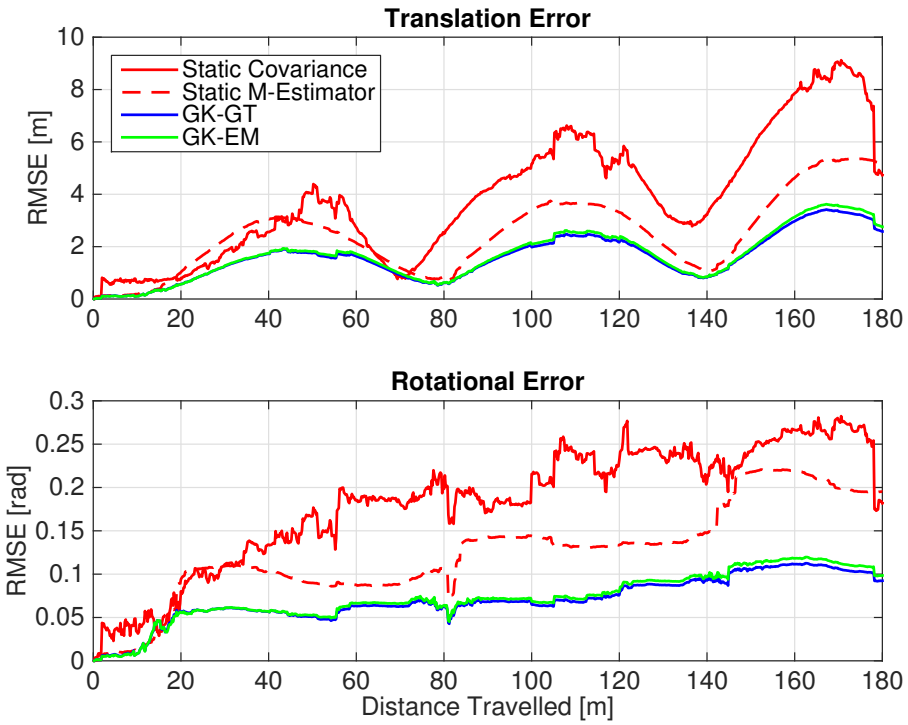

Figure 4.9: A comparison of translational and rotational Root Mean Square Error on simulated data (RMSE) for four different stereo-visual odometry pipelines: two baseline bundle adjustment procedures with and without a robust Student's t cost with a fixed and hand-tuned covariance and degrees of freedom (M-Estimation), a robust bundle adjustment with covariances learned from ground truth with algorithm 1 (GK-GT), and a robust bundle adjustment using covariances learned without ground truth using expectation maximization, with algorithm 3 (GK-EM). Note in this experiment, the RMSE curves for GK-GT and GK-EM very nearly overlap. The overall translational and rotational ARMSE values are shown in Table 4.1.

Figure 4.10: The KITTI dataset contains three different environments. We validate PROBE-GK by training on each type and testing against a baseline stereo visual odometry pipeline.

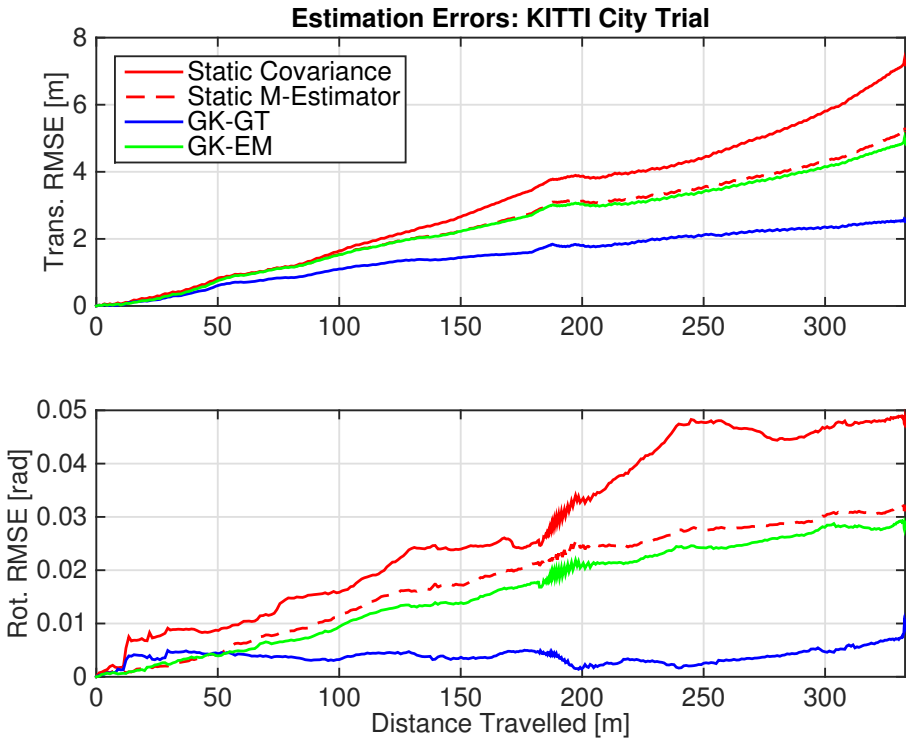

Figure 4.11: RMSE comparison of stereo odometry estimators evaluated on data from the city category in the KITTI dataset. See Table 4.1 for a quantitative summary.

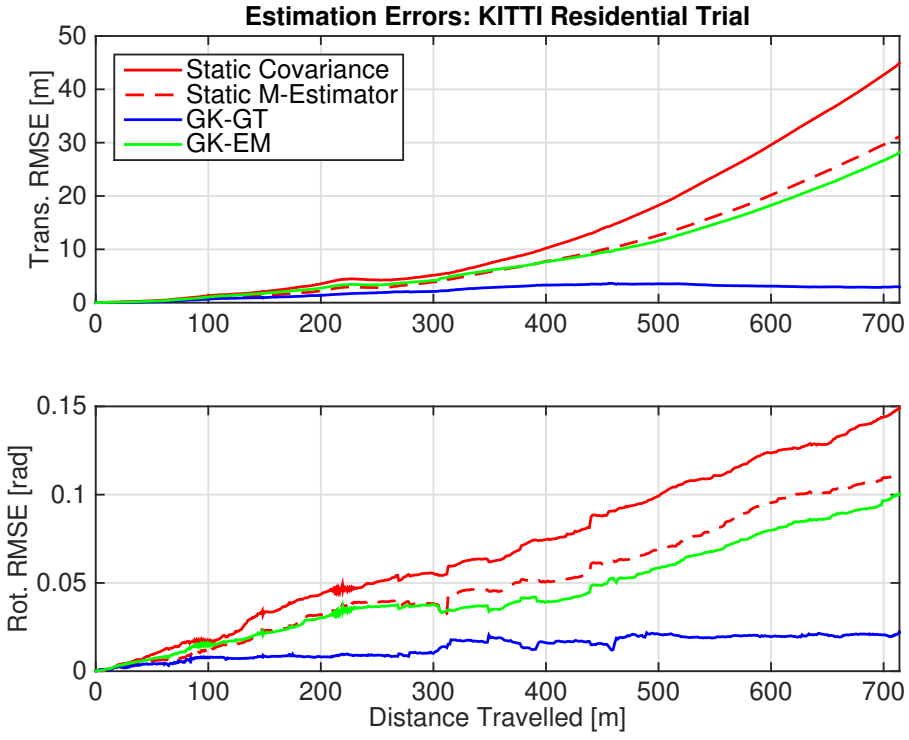

Figure 4.12: RMSE comparison of stereo odometry estimators evaluated on data from the residential category in the KITTI dataset.

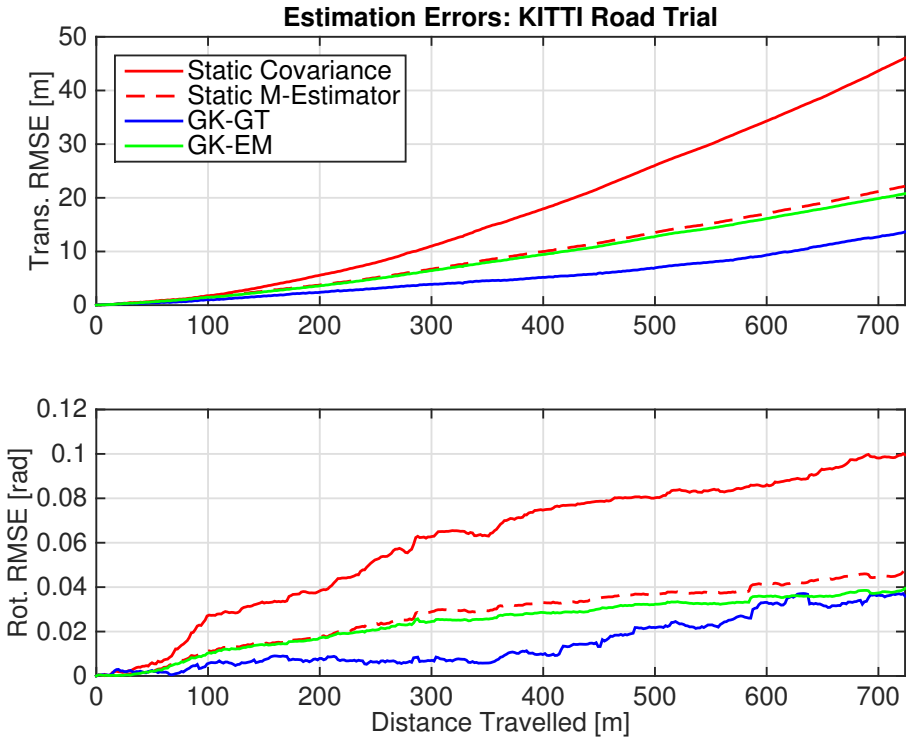

Figure 4.13: RMSE comparison of stereo odometry estimators evaluated on data from the road category in the KITTI dataset.

Table 4.1: Comparison of average root mean squared errors (ARMSE) for rotational and translational components. Each trial is trained and tested from a particular category of raw data from the synthetic and KITTI datasets.

		Trans. ARMSE [m]				Rot. ARMSE [rad]			
	Length [m]	Fixed Covar.	Static M-Estimator	GK-GT	GK-EM	Fixed Covar.	Static M-Estimator	GK-GT	GK-EM
Synthetic	180	3.87	2.49	1.59	1.66	0.18	0.13	0.070	0.073
City	332.9	3.84	2.99	1.69	2.87	0.032	0.021	0.0046	0.018
Residential	714.1	13.48	9.37	1.97	8.80	0.068	0.050	0.013	0.044
Road	723.8	17.69	9.38	5.24	8.87	0.060	0.027	0.015	0.024

Further, we observe that the performance of PROBE-GK depends on the similarity of the training data to the final test trials. A characteristic training dataset was important for consistent improvements on test trials.

4.6.3 UTIAS

To further investigate the capability of our EM approach, we evaluated PROBE-GK on experimental data collected at the University of Toronto Institute for Aerospace Studies (UTIAS). For this experiment, we drove a Clearpath Husky rover outfitted with an Ashtech DG14 Differential GPS, and a PointGrey XB3 stereo camera around the MarsDome (an indoor Mars analog testing environment) at UTIAS (Figure 4.14) for five trials of a similar path. Each trial

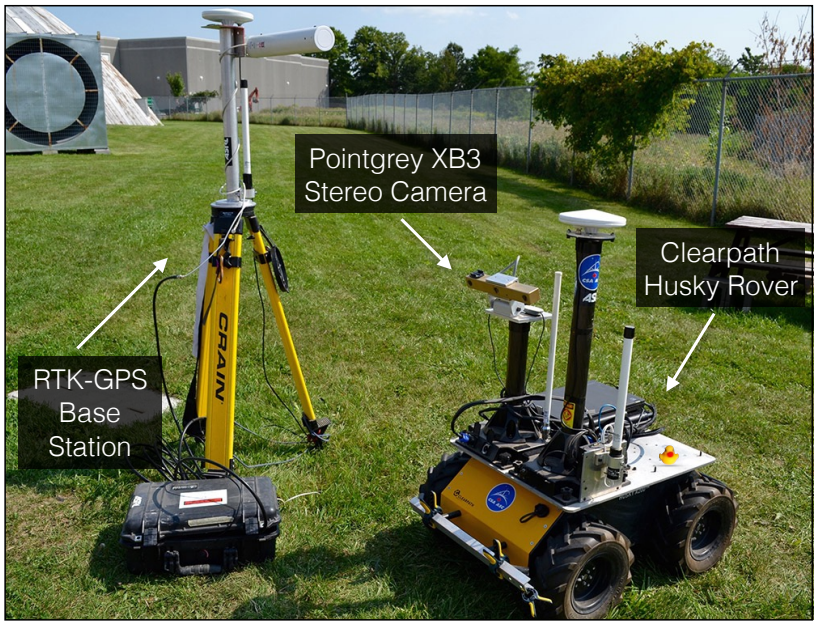

Figure 4.14: Our experimental apparatus: a Clearpath Husky rover outfitted with a PointGrey XB3 stereo camera and a differential GPS receiver and base station.

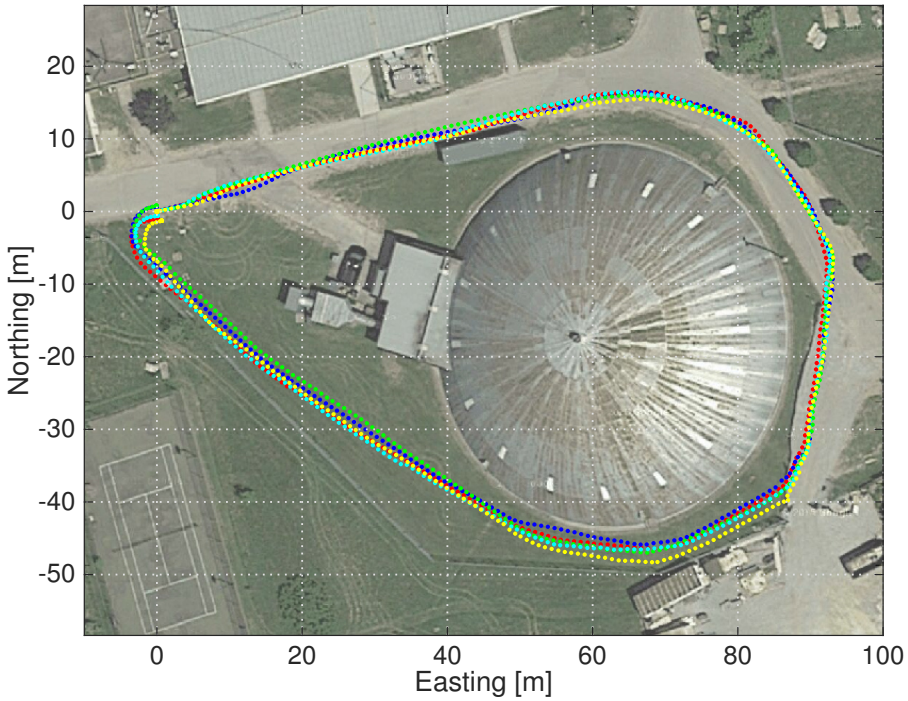

Figure 4.15: GPS ground truth for 5 experimental trials collected near the UTIAS Mars Dome. Each trial is approximately 250 m long.

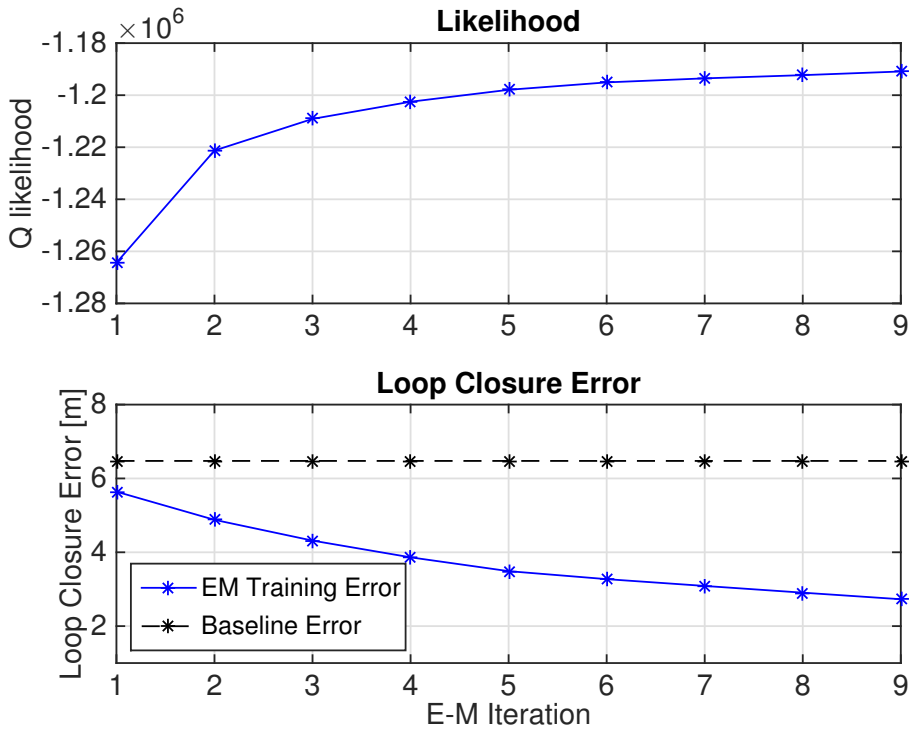

Figure 4.16: Training without ground truth using PROBE-GK-EM on a 250.2m path around the Mars Dome at UTIAS. The likelihood of the data increases with each iteration, and the loop closure error decreases, improving significantly from a baseline static M-estimator.

was approximately 250 m in length and we made an effort to align the start and end points of each loop. We used the wide baseline (25 cm) of the XB3 stereo camera to record the stereo images. The approximate trajectory for all 5 trials, as recorded by GPS, is shown in Figure 4.15. Note that the GPS data was not used during training, and only recorded for reference.

For the prediction space in our experiments, we mimicked the KITTI experiments, omitting inertial magnitudes as no inertial data was available. We trained PROBE-GK without ground truth, using the Expectation Maximization approach. Figure 4.16 shows the likelihood and loop closure error as a function of EM iteration.

The EM approach indeed produced significant error reductions on the training dataset after just a few iterations. Although it was trained with no ground truth information, our PROBE-GK model was used to produce significant reductions in the loop closure errors of the remaining 4 test trials. This reinforced our earlier hypothesis: the EM method works well when the training trajectory more closely resembles the test trials (as was the case in this experiment). Table 4.2 lists the statistics for each test.

Table 4.2: Comparison of loop closure errors for 4 different experimental trials with and without a learned PROBE-GK-EM model.

		Loop Closure Error [m]	
Trial	Path Length [m]	PROBE-GK-EM	Static M-Estimator
2	250.3	3.88	8.07
3	250.5	3.07	6.64
4	205.4	2.81	7.57
5	249.9	2.34	7.75

4.7 Summary

Predictive Robust Estimation (PROBE) applied Generalized Kernel estimation to improve on the uncorrelated and static Gaussian error models typically employed in stereo odometry. By building a non-parametric predictive model for the density of reprojection errors, we derived a robust least squares objective whose parameters were predicted based on training data. In summary, this chapter contributed

1. a probabilistic model for indirect stereo visual odometry, leading to a predictive robust algorithm for inference on that model,

2. an efficient approach to constructing the robust algorithm based on Generalized Kernel (GK) estimation,

3. a procedure for training our model using pairs of stereo images with known relative transforms, and

4. an iterative, expectation-maximization approach to train our GK model when the relative ground truth egomotion was unavailable.

Appendices

Bibliography

Agarwal, S., Mierle, K., et al. (2016). Ceres solver.

Alcantarilla, P. F. and Woodford, O. J. (2016). Noise models in feature-based stereo visual odometry.

Altmann, S. L. (1989). Hamilton, rodrigues, and the quaternion scandal. *Math. Mag.*, 62(5):291–308.

Barfoot, T. D. (2017). *State Estimation for Robotics*. Cambridge University Press.

Barfoot, T. D. and Furgale, P. T. (2014). Associating uncertainty with three-dimensional poses for use in estimation problems. *IEEE Trans. Rob.*, 30(3):679–693.

Brachmann, E. and Rother, C. (2018). Learning less is more-6d camera localization via 3d surface regression. In *Proc. CVPR*, volume 8.

Byravan, A. and Fox, D. (2017). SE3-nets: Learning rigid body motion using deep neural networks. In *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, pages 173–180.

Cadena, C., Carlone, L., Carrillo, H., Latif, Y., Scaramuzza, D., Neira, J., Reid, I., and Leonard, J. J. (2016). Past, present, and future of simultaneous localization and mapping: Toward the Robust-Perception age. *IEEE Trans. Rob.*, 32(6):1309–1332.

Carlone, L., Rosen, D. M., Calafiore, G., Leonard, J. J., and Dellaert, F. (2015a). Lagrangian duality in 3D SLAM: Verification techniques and optimal solutions. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 125–132.

Carlone, L., Tron, R., Daniilidis, K., and Dellaert, F. (2015b). Initialization techniques for 3D SLAM: A survey on rotation estimation and its use in pose graph optimization. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4597–4604.

Cheng, Y., Maimone, M. W., and Matthies, L. (2006). Visual odometry on the mars exploration rovers - a tool to ensure accurate driving and science imaging. *IEEE Robot. Automat. Mag.*, 13(2):54–62.

Clark, R., Wang, S., Wen, H., Markham, A., and Trigoni, N. (2017). Vinet: Visual-inertial odometry as a sequence-to-sequence learning problem.

Clement, L. and Kelly, J. (2018). How to train a CAT: learning canonical appearance transformations for direct visual localization under illumination change. *IEEE Robotics and Automation Letters*, 3(3):2447–2454.

Clement, L., Peretroukhin, V., and Kelly, J. (2017). Improving the accuracy of stereo visual odometry using visual illumination estimation. In Kulic, D., Nakamura, Y., Khatib, O., and Venture, G., editors, *2016 International Symposium on Experimental Robotics*, volume 1 of *Springer Proceedings in Advanced Robotics*, pages 409–419. Springer International Publishing, Berlin Heidelberg. Invited to Journal Special Issue.

Costante, G., Mancini, M., Valigi, P., and Ciarfuglia, T. A. (2016). Exploring representation learning with CNNs for Frame-to-Frame Ego-Motion estimation. *IEEE Robotics and Automation Letters*, 1(1):18–25.

Crete, F., Dolmiere, T., Ladret, P., and Nicolas, M. (2007). The blur effect: perception and estimation with a new no-reference perceptual blur metric. In *Human vision and electronic imaging XII*, volume 6492, page 64920I. International Society for Optics and Photonics.

Cvišić, I. and Petrović, I. (2015). Stereo odometry based on careful feature selection and tracking. In *Proc. European Conf. on Mobile Robots (ECMR)*, pages 1–6.

Dempster, A., Laird, N., and Rubin, D. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 1–38.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *Proc. IEEE Conf. Comput. Vision and Pattern Recognition, (CVPR)*, pages 248–255.

DeTone, D., Malisiewicz, T., and Rabinovich, A. (2016). Deep image homography estimation.

Duan, Y., Chen, X., Houthooft, R., Schulman, J., and Abbeel, P. (2016). Benchmarking deep reinforcement learning for continuous control. In *Proc. Int. Conf. on Machine Learning*, ICML'16, pages 1329–1338.

Eisenman, A. R., Liebe, C. C., and Perez, R. (2002). Sun sensing on the mars exploration rovers. In *Aerosp. Conf. Proc.*, volume 5, pages 5–2249–5–2262 vol.5. IEEE.

Engel, J., Stuckler, J., and Cremers, D. (2015). Large-scale direct SLAM with stereo cameras. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Syst. (IROS)*, pages 1935–1942.

Farnebäck, G. (2003). Two-frame motion estimation based on polynomial expansion. In *Scandinavian conference on Image analysis*, pages 363–370. Springer.

Fischler, M. and Bolles, R. (1981). Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Comm. ACM*, 24(6):381–395.

Fisher, R. (1953). Dispersion on a sphere. In *Proc. Royal Society of London A: Mathematical, Physical and Engineering Sciences*, volume 217, pages 295–305. The Royal Society.

Fitzgibbon, A. W., Robertson, D. P., Criminisi, A., Ramalingam, S., and Blake, A. (2007). Learning priors for calibrating families of stereo cameras. In *Proc. IEEE Int. Conf. Computer Vision (ICCV)*, pages 1–8.

Florez, S. A. R. (2010). *Contributions by vision systems to multi-sensor object localization and tracking for intelligent vehicles*. PhD thesis.

Forster, C., Carlone, L., Dellaert, F., and Scaramuzza, D. (2015). IMU preintegration on manifold for efficient visual-inertial maximum-a-posteriori estimation.

Forster, C., Pizzoli, M., and Scaramuzza, D. (2014). SVO: Fast semi-direct monocular visual odometry. In *Proc. IEEE Int. Conf. Robot. Automat.(ICRA)*, pages 15–22. IEEE.

Furgale, P. (2011). *Extensions to the Visual Odometry Pipeline for the Exploration of Planetary Surfaces*. PhD thesis.

Furgale, P. and Barfoot, T. D. (2010). Visual teach and repeat for long-range rover autonomy. *J. Field Robot.*, 27(5):534–560.

Furgale, P., Carle, P., Enright, J., and Barfoot, T. D. (2012). The devon island rover navigation dataset. *Int. J. Rob. Res.*, 31(6):707–713.

Furgale, P., Enright, J., and Barfoot, T. (2011). Sun sensor navigation for planetary rovers: Theory and field testing. *IEEE Trans. Aerosp. Electron. Syst.*, 47(3):1631–1647.

Furgale, P., Rehder, J., and Siegwart, R. (2013). Unified temporal and spatial calibration for multi-sensor systems. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1280–1286.

Gal, Y. (2016). *Uncertainty in Deep Learning*. PhD thesis, University of Cambridge.

Gal, Y. and Ghahramani, Z. (2016a). Bayesian convolutional neural networks with Bernoulli approximate variational inference. In *Proc. Int. Conf. Learning Representations (ICLR), Workshop Track*.

Gal, Y. and Ghahramani, Z. (2016b). Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *Proc. Int. Conf. Mach. Learning (ICML)*, pages 1050–1059.

Garg, R., Carneiro, G., and Reid, I. (2016). Unsupervised CNN for single view depth estimation: Geometry to the rescue. In *European Conf. on Comp. Vision*, pages 740–756. Springer.

Geiger, A., Lenz, P., Stiller, C., and Urtasun, R. (2013). Vision meets robotics: The KITTI dataset. *Int. J. Rob. Res.*, 32(11):1231–1237.

Geiger, A., Ziegler, J., and Stiller, C. (2011). StereoScan: Dense 3D reconstruction in real-time. In *Proc. IEEE Intelligent Vehicles Symp. (IV)*, pages 963–968.

Geman, S., McClure, D. E., and Geman, D. (1992). A nonlinear filter for film restoration and other problems in image processing. *CVGIP: Graphical models and image processing*, 54(4):281–289.

Glocker, B., Izadi, S., Shotton, J., and Criminisi, A. (2013). Real-time rgb-d camera relocalization. In *2013 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 173–179.

Grewal, M. S. and Andrews, A. P. (2010). Applications of kalman filtering in aerospace 1960 to the present [historical perspectives]. *IEEE Control Syst. Mag.*, 30(3):69–78.

Haarnoja, T., Ajay, A., Levine, S., and Abbeel, P. (2016). Backprop KF: Learning discriminative deterministic state estimators. In *Proc. Advances in Neural Inform. Process. Syst. (NIPS)*.

Handa, A., Bloesch, M., Pătrăucean, V., Stent, S., McCormac, J., and Davison, A. (2016). gvnn: Neural network library for geometric computer vision. In *Computer Vision – ECCV 2016 Workshops*, pages 67–82. Springer, Cham.

Hartley, R., Trumpf, J., Dai, Y., and Li, H. (2013). Rotation averaging. *Int. J. Comput. Vis.*, 103(3):267–305.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Hu, H. and Kantor, G. (2015). Parametric covariance prediction for heteroscedastic noise. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Syst. (IROS)*, pages 3052–3057.

Huber, P. J. (1964). Robust estimation of a location parameter. *The Annals of Mathematical Statistics*, pages 73–101.

Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., and Darrell, T. (2014). Caffe: Convolutional architecture for fast feature embedding. In *Proc. ACM Int. Conf. Multimedia (MM)*, pages 675–678.

Kelly, J., Saripalli, S., and Sukhatme, G. S. (2008). Combined visual and inertial navigation for an unmanned aerial vehicle. In *Proc. Field and Service Robot. (FSR)*, pages 255–264.

Kendall, A. and Cipolla, R. (2016). Modelling uncertainty in deep learning for camera relocalization. In *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, pages 4762–4769.

Kendall, A. and Cipolla, R. (2017). Geometric loss functions for camera pose regression with deep learning. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6555–6564.

Kendall, A., Grimes, M., and Cipolla, R. (2015). PoseNet: A convolutional network for Real-Time 6-DOF camera relocalization. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 2938–2946.

Kerl, C., Sturm, J., and Cremers, D. (2013). Robust odometry estimation for RGB-D cameras. In *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, pages 3748–3754.

Lakshminarayanan, B., Pritzel, A., and Blundell, C. (2017). Simple and scalable predictive uncertainty estimation using deep ensembles. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems 30*, pages 6402–6413. Curran Associates, Inc.

Lalonde, J.-F., Efros, A. A., and Narasimhan, S. G. (2011). Estimating the natural illumination conditions from a single outdoor image. *Int. J. Comput. Vis.*, 98(2):123–145.

Lambert, A., Furgale, P., Barfoot, T. D., and Enright, J. (2012). Field testing of visual odometry aided by a sun sensor and inclinometer. *J. Field Robot.*, 29(3):426–444.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521(7553):436–444.

Lee, S., Purushwalkam, S., Cogswell, M., Crandall, D., and Batra, D. (2015). Why M heads are better than one: Training a diverse ensemble of deep networks.

Leutenegger, S., Lynen, S., Bosse, M., Siegwart, R., and Furgale, P. (2015). Keyframe-based visual–inertial odometry using nonlinear optimization. *Int. J. Rob. Res.*, 34(3):314–334.

Levine, S., Finn, C., Darrell, T., and Abbeel, P. (2016). End-to-end training of deep visuomotor policies. *J. Mach. Learn. Res.*

Li, Q., Qian, J., Zhu, Z., Bao, X., Helwa, M. K., and Schoellig, A. P. (2017a). Deep neural networks for improved, impromptu trajectory tracking of quadrotors. In *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, pages 5183–5189.

Li, R., Wang, S., Long, Z., and Gu, D. (2017b). UnDeepVO: Monocular visual odometry through unsupervised deep learning.

Lucas, B. D. and Kanade, T. (1981). An iterative image registration technique with an application to stereo vision. In *Proceedings of the 7th International Joint Conference on Artificial Intelligence - Volume 2*, IJCAI'81, pages 674–679, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Ma, W.-C., Wang, S., Brubaker, M. A., Fidler, S., and Urtasun, R. (2016). Find your way by observing the sun and other semantic cues.

MacTavish, K. and Barfoot, T. D. (2015). At all costs: A comparison of robust cost functions for camera correspondence outliers. In *Proc. Conf. on Comp. and Robot Vision (CRV)*, pages 62–69.

Maddern, W., Pascoe, G., Linegar, C., and Newman, P. (2016). 1 year, 1000 km: The oxford RobotCar dataset. *Int. J. Rob. Res.*

Maimone, M., Cheng, Y., and Matthies, L. (2007). Two years of visual odometry on the mars exploration rovers. *J. Field Robot.*, 24(3):169–186.

Mayor, A. (2019). *Gods and Robots.* Princeton University Press.

McManus, C., Upcroft, B., and Newman, P. (2014). Scene signatures: Localised and point-less features for localisation. In *Proc. Robotics: Science and Systems X.*

Melekhov, I., Ylioinas, J., Kannala, J., and Rahtu, E. (2017). Relative camera pose estimation using convolutional neural networks. In *Proc. Int. Conf. on Advanced Concepts for Intel. Vision Syst.*, pages 675–687. Springer.

Nilsson, N. J. (1984). Shakey the robot. Technical report, SRI International.

Oliveira, G. L., Radwan, N., Burgard, W., and Brox, T. (2017). Topometric localization with deep learning. *arXiv preprint arXiv:1706.08775*.

Olson, C. F., Matthies, L. H., Schoppers, M., and Maimone, M. W. (2003). Rover navigation using stereo ego-motion. *Robot. Auton. Syst.*, 43(4):215–229.

Osband, I., Blundell, C., Pritzel, A., and Van Roy, B. (2016). Deep exploration via bootstrapped DQN. In *Proc. Advances in Neural Inform. Process. Syst. (NIPS)*, pages 4026–4034.

Peretroukhin, V., Clement, L., Giamou, M., and Kelly, J. (2015a). PROBE: Predictive robust estimation for visual-inertial navigation. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'15)*, pages 3668–3675, Hamburg, Germany.

Peretroukhin, V., Clement, L., and Kelly, J. (2015b). Get to the point: Active covariance scaling for feature tracking through motion blur. In *Proceedings of the IEEE International Conference on Robotics and Automation Workshop on Scaling Up Active Perception*, Seattle, Washington, USA.

Peretroukhin, V., Clement, L., and Kelly, J. (2017). Reducing drift in visual odometry by inferring sun direction using a bayesian convolutional neural network. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'17)*, pages 2035–2042, Singapore.

Peretroukhin, V., Clement, L., and Kelly, J. (2018). Inferring sun direction to improve visual odometry: A deep learning approach. *International Journal of Robotics Research*, 37(9):996–1016.

Peretroukhin, V. and Kelly, J. (2018). DPC-Net: Deep pose correction for visual localization. *IEEE Robotics and Automation Letters*, 3(3):2424–2431.

Peretroukhin, V., Kelly, J., and Barfoot, T. D. (2014). Optimizing camera perspective for stereo visual odometry. In *Canadian Conference on Comp. and Robot Vision*, pages 1–7.

Peretroukhin, V., Vega-Brown, W., Roy, N., and Kelly, J. (2016). PROBE-GK: Predictive robust estimation using generalized kernels. In *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, pages 817–824.

Peretroukhin, V., Wagstaff, B., and Kelly, J. (2019). Deep probabilistic regression of elements of SO(3) using quaternion averaging and uncertainty injection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'19) Workshop on Uncertainty and Robustness in Deep Visual Learning*, pages 83–86, Long Beach, California, USA.

Punjani, A. and Abbeel, P. (2015). Deep learning helicopter dynamics models. In *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, pages 3223–3230.

Redfield, S. (2019). A definition for robotics as an academic discipline. *Nature Machine Intelligence*, 1(6):263–264.

Rosen, D. M., Carlone, L., Bandeira, A. S., and Leonard, J. J. (2019). SE-Sync: A certifiably correct algorithm for synchronization over the special euclidean group. *Int. J. Rob. Res.*, 38(2-3):95–125.

Scaramuzza, D. and Fraundorfer, F. (2011). Visual odometry [tutorial]. *IEEE Robot. Autom. Mag.*, 18(4):80–92.

Sibley, G., Matthies, L., and Sukhatme, G. (2007). Bias reduction and filter convergence for long range stereo. In *Robotics Research*, pages 285–294. Springer Berlin Heidelberg.

Sola, J. (2017). Quaternion kinematics for the error-state kalman filter. *arXiv preprint arXiv:1711.02508*.

Solà, J., Deray, J., and Atchuthan, D. (2018). A micro lie theory for state estimation in robotics.

Sünderhauf, N., Shirazi, S., Dayoub, F., Upcroft, B., and Milford, M. (2015). On the performance of ConvNet features for place recognition. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Syst. (IROS)*, pages 4297–4304.

Sunderhauf, N., Shirazi, S., Jacobson, A., Dayoub, F., Pepperell, E., Upcroft, B., and Milford, M. (2015). Place recognition with ConvNet landmarks: Viewpoint-robust, condition-robust, training-free. In *Proc. Robotics: Science and Systems XII*.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015). Going deeper with convolutions. In *Proc. IEEE Conf. Comput. Vision and Pattern Recognition, (CVPR)*, pages 1–9.

Thrun, S., Montemerlo, M., Dahlkamp, H., Stavens, D., Aron, A., Diebel, J., Fong, P., Gale, J., Halpenny, M., Hoffmann, G., Lau, K., Oakley, C., Palatucci, M., Pratt, V., Stang, P., Strohband, S., Dupont, C., Jendrossek, L.-E., Koelen, C., Markey, C., Rummel, C., van Niekerk,

J., Jensen, E., Alessandrini, P., Bradski, G., Davies, B., Ettinger, S., Kaehler, A., Nefian, A., and Mahoney, P. (2006). Stanley: The robot that won the DARPA grand challenge. *J. Field Robotics*, 23(9):661–692.

Tsotsos, K., Chiuso, A., and Soatto, S. (2015). Robust inference for visual-inertial sensor fusion. In *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, pages 5203–5210.

Umeyama, S. (1991). Least-Squares estimation of transformation parameters between two point patterns. *IEEE Trans. Pattern Anal. Mach. Intell.*, 13(4):376–380.

Vega-Brown, W. and Roy, N. (2013). CELLO-EM: Adaptive sensor models without ground truth. In *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, pages 1907–1914.

Vega-Brown, W. R., Doniec, M., and Roy, N. G. (2014). Nonparametric Bayesian inference on multivariate exponential families. In *Proc. Advances in Neural Information Proc. Syst. (NIPS) 27*, pages 2546–2554.

Wang, S., Clark, R., Wen, H., and Trigoni, N. (2017). DeepVO: Towards end-to-end visual odometry with deep recurrent convolutional neural networks. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2043–2050.

Yang, F., Choi, W., and Lin, Y. (2016). Exploit all the layers: Fast and accurate cnn object detector with scale dependent pooling and cascaded rejection classifiers. In *Proc. IEEE Int. Conf. Comp. Vision and Pattern Recognition (CVPR)*, pages 2129–2137.

Yang, N., Wang, R., Stueckler, J., and Cremers, D. (2018). Deep virtual stereo odometry: Leveraging deep depth prediction for monocular direct sparse odometry. In *European Conference on Computer Vision (ECCV)*. accepted as oral presentation, arXiv 1807.02570.

Zhang, G. and Vela, P. (2015). Optimally observable and minimal cardinality monocular SLAM. In *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, pages 5211–5218.

Zhou, B., Krähenbühl, P., and Koltun, V. (2019). Does computer vision matter for action?

Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., and Oliva, A. (2014). Learning deep features for scene recognition using places database. In *Advances in Neural Inform. Process. Syst. (NIPS)*, pages 487–495.

Zhou, T., Brown, M., Snavely, N., and Lowe, D. G. (2017). Unsupervised learning of depth and Ego-Motion from video. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6612–6619.