Visual Odometry Aided by a Sun Sensor and an Inclinometer

by

Andrew J. Lambert

A thesis submitted in conformity with the requirements
for the degree of Master of Applied Science
Graduate Department of Aerospace Science and Engineering
University of Toronto

Abstract

Visual Odometry Aided by a Sun Sensor and an Inclinometer

Andrew J. Lambert

Master of Applied Science

Graduate Department of Aerospace Science and Engineering

University of Toronto

2011

Due to the absence of any satellite-based global positioning system on Mars, the Mars Exploration Rovers commonly track position changes of the vehicle using a technique called visual odometry (VO), where updated rover poses are determined by tracking keypoints between stereo image pairs. Unfortunately, the error of VO grows super-linearly with the distance traveled, primarily due to the contribution of orientation error. This thesis outlines a novel approach incorporating sun sensor and inclinometer measurements directly into the VO pipeline, utilizing absolute orientation information to reduce the error growth of the motion estimate. These additional measurements have very low computation, power, and mass requirements, providing a localization improvement at nearly negligible cost. The mathematical formulation of this approach is described in detail, and extensive results are presented from experimental trials utilizing data collected during a 10 kilometre traversal of a Mars analogue site on Devon Island in the Canadian High Arctic.

Acknowledgements

This thesis could not have been completed without the expert guidance and mathematical wizardry of my supervisor, Dr. Tim Barfoot.

Thanks to Paul Furgale for patiently sharing his boundless technical expertise with me (and for pushing that cart all over Devon Island).

To all the members of the Autonomous Space Robotics Lab, each of whom helped me in countless ways, I am truly grateful.

Tremendous thanks to Erin Franklin, for supporting me every step of the way and providing creative consultation on all my work.

Finally, thanks to my parents, Rick and Debbie, for helping me through all the years of school that built up to the completion of this thesis.

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.
$E[\cdot]$	:	The expectation operator.
$\underrightarrow{\mathcal{F}}_a$	:	A reference frame in three dimensions.
$\bar{(\cdot)}$	:	Symbols with an overbar are nominal values of a quantity.
$\hat{(\cdot)}$	:	Symbols with a hat are estimates of a true quantity.
$\tilde{(\cdot)}$	:	Symbols with a tilde are measured quantities.
$(\cdot)^{\times}$	:	The cross-product operator that produces a skew-symmetric matrix from a 3×1 column.
$\mathbf{1}$	:	The identity matrix.
$\mathbf{0}$	:	The zero matrix.
$\boldsymbol{\rho}_a^{d,e}$	:	A vector from point e to point d (denoted by the superscript) and expressed in $\underrightarrow{\mathcal{F}}_a$ (denoted by the subscript).
$\mathbf{C}_{a,b}$	:	The 3×3 rotation matrix that transforms vectors from $\underrightarrow{\mathcal{F}}_b$ to $\underrightarrow{\mathcal{F}}_a$.

Chapter 1

Introduction

Since landing in 2004, the Mars Exploration Rovers (MERs) have been traversing the Martian surface in search of and gathering scientific data. Due to communication latency and bandwidth limitations, the rovers are usually commanded once per Martian sol using a prescheduled sequence of driving directions (Matthies et al., 2007). Accurate autonomous execution of these commands maximizes the scientific yield of the mission and allows the rover to avoid potentially dangerous locations. Thus, accurate vehicle localization is a critical aspect of the planetary rover operations on Mars.

During autonomous traverses across nominal terrain, the MERs use wheel odometry measurements to track position changes of the vehicle, due to the absence of any satellite-based global positioning system (GPS) on Mars. However, the rovers have frequently encountered steep slopes and sandy terrain, resulting in large amounts of wheel slip and rendering the odometry measurements unreliable. Figure 1.1 depicts a particularly challenging example from the Opportunity rover, in which the vehicle rotated its wheels enough to have travelled 50 meters, but was only displaced by 2 meters due to extensive wheel slip.

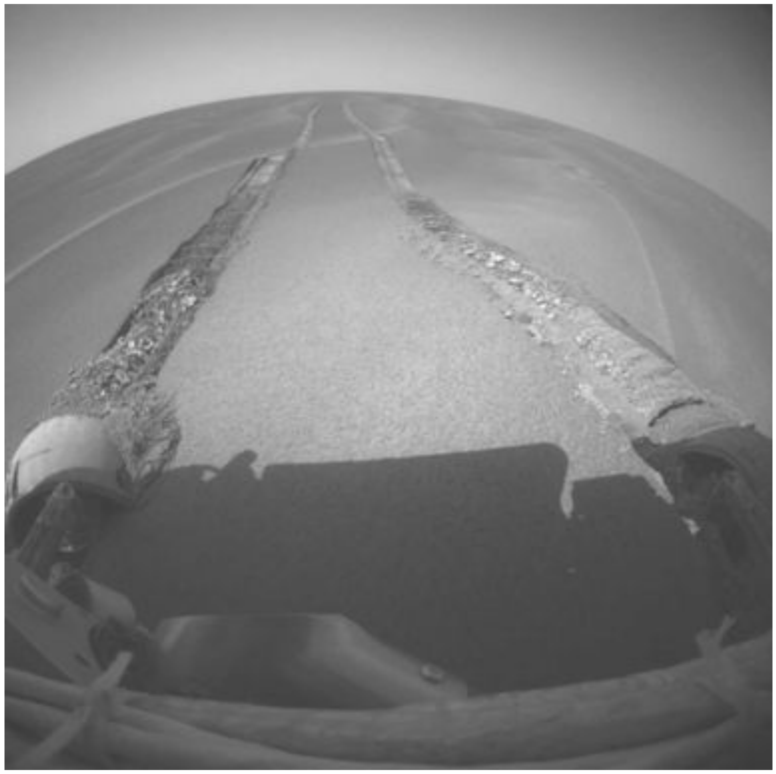


Figure 1.1: A photograph from the Opportunity Rover at Purgatory Ripple on Mars, depiciting extensive wheel slip (Maimone et al., 2007). Photo credit: JPL/Caltech.

For these high-slip situations, the MERs employ a technique called *visual odometry* (VO), where updated vehicle poses are determined by tracking keypoints between stereo image pairs. This method can provide accurate localization in cases of wheel slip or wheel dragging, but can also act as an initial slip detection tool to increase vehicle safety (Maimone et al., 2007). The use of VO is currently limited to short drives due to computation time, but in future planetary missions with increased processor power, the benefits of visual odometry will be useful over longer range traverses. Unfortunately, the error of VO grows super-linearly with the distance travelled, primarily due to the contribution of orientation error (Olson et al., 2003).

This thesis outlines a novel approach that has been developed to counteract this phenomenon, incorporating sun sensor and inclinometer measurements directly into the VO pipeline to limit error growth of the motion estimate. This approach was motivated by previous research demonstrating that periodic absolute orientation updates, such as those obtained from a sun sensor, restrict the error to grow only linearly with distance (Olson et al., 2003). Note that the use of a sun sensor (or similarly a startracker) is well-suited for the exploration of Mars (or the Moon), whose lack of a useful magnetic field precludes the use of a compass for absolute heading information (Eisenman et al., 2002). Additionally, a sun sensor is not subject to drift, which can plague inertial sensors used to monitor heading over long distances and times. The novel technique detailed in this thesis builds upon this periodic approach, incorporating sun sensor and inclinometer measurements directly into the visual odometry solution as they are acquired. This formulation allows for continuous correction of the vehicle's heading estimate, allowing for greatly improved accuracy over long distances. The sun sensor provides absolute heading information (as well as pitch/roll over long periods of time), while the inclinometer measures the pitch and roll of the rover platform, allowing the application of angular corrections to the full attitude of the vehicle.

This thesis is organized as follows. Chapter 2 outlines the major elements of the visual odometry pipeline and examines the relevant techniques that have been employed in previous research. Prior work regarding the use of sun sensors in rover navigation is also reviewed. Chapter 3 outlines the mathematical formulation of our estimation framework. Chapter 4 begins by providing the details of our experimental test data acquired at a Mars analog site on Devon Island in the Canadian Arctic. Next, detailed results are presented from the experimental trials. This includes an evaluation of stereo feature detectors, an demonstration of how the sun sensor and inclinometer measurements greatly improve visual odometry estimates, and an examination of the computational gains made possible by this algorithm. Chapter 5 provides a thorough discussion of the inherent VO bias for planetary rovers that was discovered during the experimental trials. Chapter 6 summarizes the findings of this thesis, and looks forward to possible extensions of this work in future research.

Chapter 2

Visual Odometry and Sun Sensing for Rover Navigation

The original visual odometry system was introduced by Moravec (1980) in his doctoral thesis. The implementation was simple, but it was the first to contain the fundamental elements of the modern VO pipeline: a keypoint detector, a keypoint matching algorithm, and a motion estimator. This basic model was continued by Matthies, who achieved significantly better accuracy by treating the motion estimation as a statistical estimation problem, and modelling the landmark uncertainties as ellipsoidal three-dimensional Gaussians (Matthies and Shafer, 1987; Matthies, 1992). The system outlined by Matthies formed the basis of the MER VO algorithm (Matthies et al., 2007), and helped to established the modern visual odometry pipeline, as illustrated in Figure 2.1.

This chapter will describe in detail the major elements of this pipeline and examine the relevant techniques that have been employed in previous research. Prior work regarding the use of sun sensors in rover navigation will also reviewed.

.

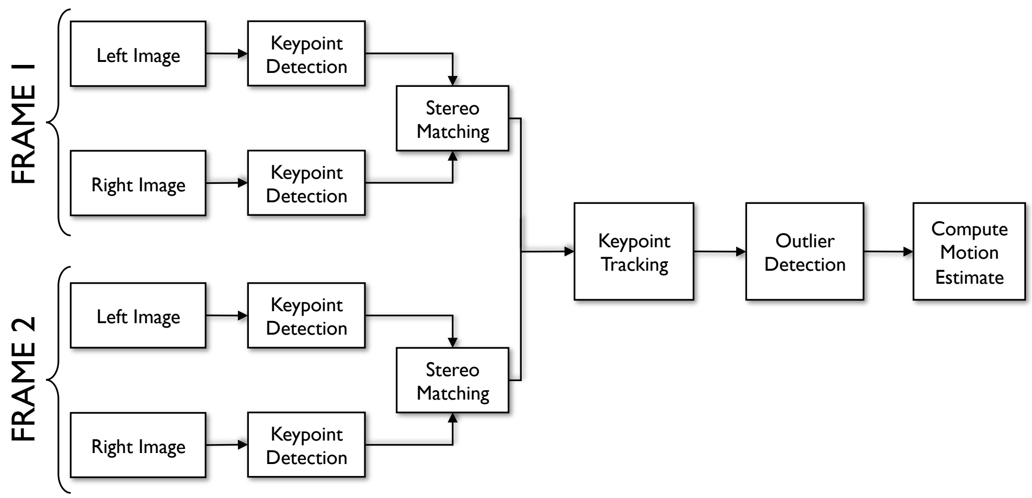


Figure 2.1: The basic visual odometry pipleline using a stereo camera.

2.1 The Visual Odometry Pipeline

2.1.1 Keypoint Detection

Once a stereo image has been acquired from the camera, rectified, and processed into a usable form, the first step in the visual odometry pipeline is to extract distinct points of interest, or *keypoints*, from within the image. Figure 2.1.1 illustrates the large number of keypoints that are typically detected within a single image. A number of different keypoint detectors have been employed in the literature for this task, each with its own unique properties. The Mars Exploration Rovers use the Harris corner detector, which uses the auto-correlation function to measure changes in image intensity that result from shifting the image patch (Harris and Stephens, 1988). Harris corners are simple to compute, and thus are well-suited for computationally limited planetary exploration rovers. Another computationally efficient method is the FAST (Features from Accelerated Segment Test) feature detector (Rosten and Drummond, 2005, 2006). In this algorithm, a circle of pixels surrounding a candidate point is examined, and a feature is detected if a certain number of these contiguous pixels are above or below the intensity of the candidate by some threshold. Because of its high speed and reliability, the FAST detector has been used extensively in the literature (Mei et al., 2010; Howard, 2008; Konolige and Bowman,

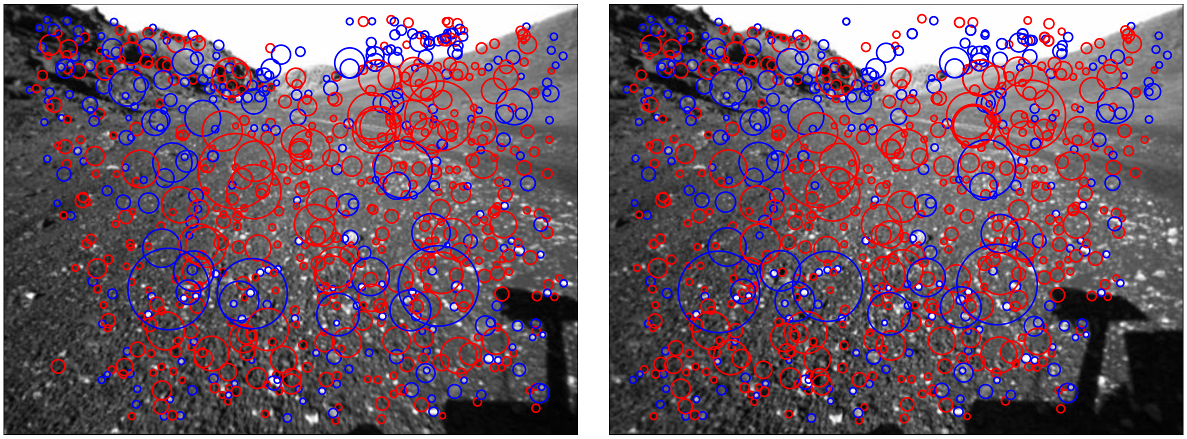


Figure 2.2: Keypoints extracted from the left and right images of a stereo pair using the SURF detector. The sizes of the circles indicate the scale of the keypoint that was detected. The blue circles represent light blobs on dark backgrounds, while red corresponds to dark blobs on light backgrounds.

2009). Another option is the SURF (Speeded-Up Robust Features) detector (Bay et al., 2008), which uses integral images to detect blob-like structures at multiple scales. The technique is based on the Scale Invariant Feature Transform (SIFT) algorithm (Lowe, 2004), but uses a number of fast approximations to speed up performance. A GPU implementation of the SURF algorithm was recently utilized by Furgale and Barfoot (2010) for a visual-teach-and-repeat system. In Chapter 4, the Harris, FAST, and SURF keypoint detectors are compared experimentally to determine which detector produces the most accurate visual odometry estimate as compared to groundtruth.

2.1.2 Stereo Matching

After keypoint detection has been performed, the next step in the VO pipeline is to match similar keypoints between the left and right images of the stereo pair. Once a keypoint pair has been established, the pixel location disparity of the keypoint in the left and right images can be measured, as shown in Figure 2.3. Chapter 3 details how this disparity measurement can be used to triangulate the three-dimensional location of the observed landmark, given the camera baseline and focal length.

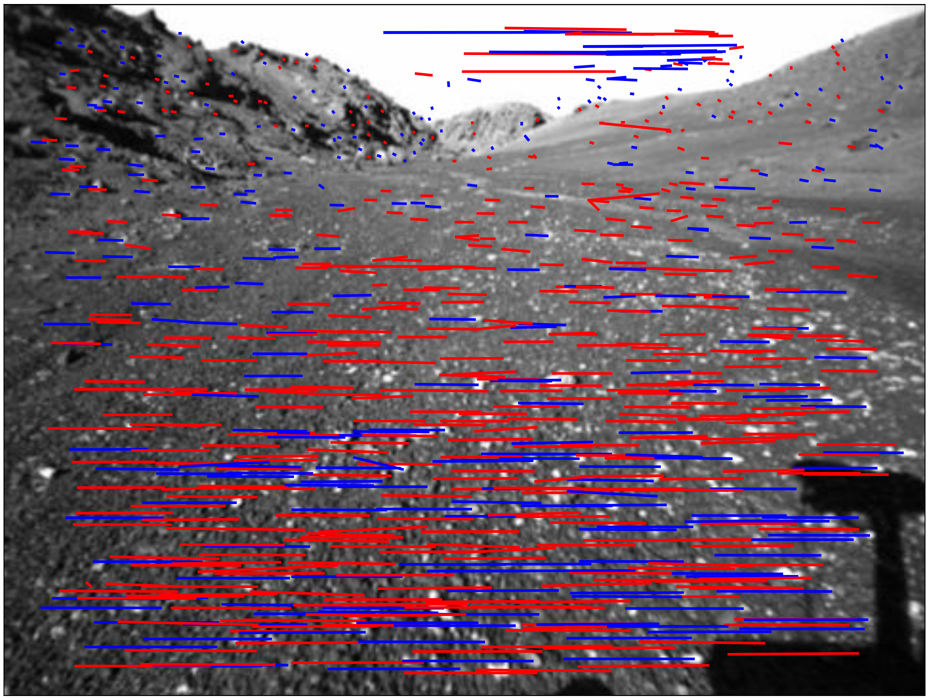
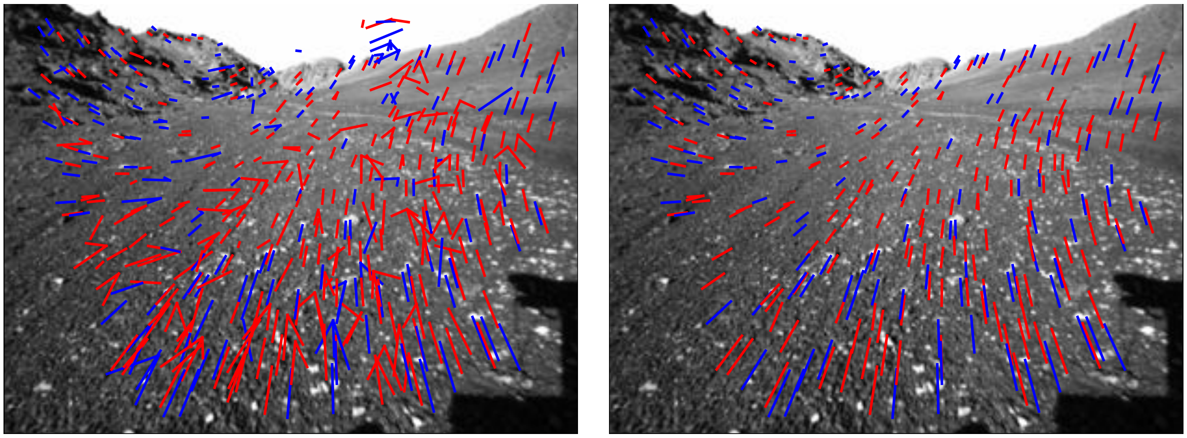


Figure 2.3: Disparity measurements produced by matching the keypoints observed in Figure 2.1.1. Note that matches in the foreground generally have large disparity, indicating these keypoints are close, while those in the background have small disparity, indicating they are far. There are some mismatches as well, such as those in the sky portion of the image.

If the images have not been rectified, a two-dimensional search must be performed within the image to find a corresponding keypoint. However, image rectification aligns the left and right images of the stereo pair, such that a point in space projects onto the same row of each image. Thus, in this thesis, keypoints are detected in the left and right images, and matches are found by searching along the epipolar line of the rectified stereo pair, plus or minus some allowable tolerance to account for noise. This is a commonly used approach in the literature (Mei et al., 2010; Nister et al., 2006), but an alternative technique is to detect keypoints in a single image, and then search for correspondences using dense stereo processing (Howard, 2008; Konolige et al., 2007). On the slow processors of the MERs, down-sampled image pyramids are used to speed up correlation (Johnson et al., 2008). In most VO systems, keypoint similarity is computed using a normalized cross correlation approach (Maimone et al., 2007; Konolige et al., 2007).



(a) Raw feature tracks, with remaining outliers. (b) Feature tracks after outlier rejection.

Figure 2.4: During the feature matching and tracking stages, some erroneous correspondences will be made, resulting in outlying feature tracks. These outliers are clearly apparent in (a), where many of the feature tracks seem to contradict the consensus motion. To produce the feature tracks observed in (b), outliers were removed using a RANSAC technique, resulting in a set of tracks that support a single motion hypothesis.

2.1.3 Keypoint Tracking

After matching stereo keypoints in the first frame, the rover may drive some distance and then acquire a new stereo image. Keypoints will then be detected and matched in this new stereo frame, following the same procedures outlined above. After this has been completed, the next task in the stereo pipeline is to temporally match the keypoints across these two stereo frames, producing the feature tracks observed in Figure 2.4(a). By estimating the relative movement of the these tracked landmarks in the two stereo frames, an estimate of the rover's motion can be computed.

Keypoint tracking is similar in principle to the keypoint matching described above, but the search area is no longer defined by epipolar lines; some idea of the rover's motion between the two stereo frames must be used to constrain the window in which the corresponding keypoint may exist. Some simple approaches are to assume that the rover motion is small (Nister et al., 2006) or is at constant velocity (Davison et al., 2007), which can work if VO is running at a high framerate. A more informed technique is to use the coarse motion estimate from an IMU (Konolige et al., 2007). On the MERs, IMU and wheel odometry measurements are used to predict the attitude and position changes of the rover, respectively (Johnson et al., 2008). This approach has been upgraded for the MSL rover, where predictions are made using conservative motion bounds that account for potential wheel slip (Johnson et al., 2008).

2.1.4 Outlier Detection

A significant limitation of the keypoint matching and tracking algorithms outlined above is that they are purely appearance based; essentially, a correspondence occurs when one image patch is found that looks fairly similar to another image patch. This approach, while simple and computationally efficient, inevitably leads to mismatches and outlying feature tracks. This is especially true in terrain with repetitive patterns and texture, such as the sand and rock fields observed on Mars. Outliers can clearly be observed in Figure 2.4(a), where many of the feature tracks appear to contradict the consensus motion. Removal of these outliers is critical for obtaining accurate motion estimates from visual odometry.

The most common technique for outlier removal in visual odometry is an iterative process known as Random Sample and Consensus, or RANSAC (Fischler and Bolles, 1981). In each iteration of RANSAC, a model of the data is generated using a randomly selected minimal subset. Next, this model is evaluated by tallying how many of the remaining data points fit the model within a fixed threshold. After a set number of iterations, the most successful model is re-estimated using all its inlying points, producing a coarse motion estimate that can be used to initialize the nonlinear numerical solution of bundle adjustment. While the standard RANSAC approach has been used extensively in the literature (Maimone et al., 2007; Konolige et al., 2007; Mei et al., 2010), a number of variations have been developed to speed computation. One such flavour is preemptive RANSAC (Nister, 2003), where a set number of models are generated up front. These models are then iteratively evaluated using small subsets of the data, and the most unlikely hypotheses are rejected. This reduces the computational effort significantly, as many of the models are eliminated using a small number of points instead of the entire set. Preemptive RANSAC was used to generate the inlying feature tracks shown in Figure 2.4(b), as well as in the experiments of Chapter 4.

2.1.5 Compute Motion Estimate

After removing outlying feature tracks, the final step in the visual odometry pipeline is to compute a maximum likelihood estimate for camera translation and rotation between the acquisition of the two stereo frames. As previously mentioned, Matthies and Shafer (1987) established the probabilistic foundation for solving this problem, modelling the landmark uncertainties as ellipsoidal three-dimensional Gaussians and using a covariance-weighted nonlinear least-squares approach to solve for the rover pose update. This algorithm was used successfully on the MER rovers with only minor changes from the system originally described in Matthies' doctoral thesis (Maimone et al., 2007). However, most of the notable contemporary VO systems (Nister et al., 2006; Konolige et al., 2007; Mei et al., 2010) utilize a technique called bundle adjustment (Brown, 1958), an iterative Gauss-Newton minimization algorithm that solves for both motion and structure; that is to say, it solves for both the rover pose and the three-dimensional locations of the landmarks (although the landmark positions are not strictly required in VO). Chapter 3 explains in detail the frame-to-frame bundle adjustment solution and how it can be solved in a computationally efficient manner by exploiting the sparsity patterns of the matrices involved.

One of the inherent benefits of the bundle adjustment framework is that it allows for multi-frame solutions; in other words, one can utilize the measurements from a sliding window of frames to improve motion estimate accuracy. This approach has been used to great effect in a number of systems (Mei et al., 2010; Li et al., 2007); in particular, Konolige et al. (2007) used a sliding window bundle adjustment formulation, fused with IMU measurements using an EKF, to achieve a max error of 0.08% over a 9 km traverse. However, incorporating the additional measurements from a handful of frames significantly increases the computational burden of bundle adjustment; as the goal of this thesis was to improve accuracy while having a comparable computational cost to the MERs, a frame-to-frame approach has been used in all experiments.

2.2 Sun Sensing in Rover Navigation

The inclusion of a sun sensor on future rover missions was one of the recommendations made after the 1997 Mars Pathfinder mission (Wilcox and Nguyen, 1998). Subsequently, there were several studies at NASA's Jet Propulsion Lab into the use of a dedicated sun sensor for rover navigation. A visual odometry simulation study by Olson et al. (2003) found that long range position error grows super-linearly with the distance travelled, predominantly due to the contribution of orientation error. However, if absolute orientation measurements are available, the error is bounded to grow linearly with distance. Volpe (1999) performed field tests using the Rocky 7 rover, in which wheel odometry motion estimates were corrected using a sun sensor, in conjunction with an accelerometer to determine sensor tilt. These tests experimentally verified that the use of an absolute orientation sensor can restrict the positional error of a rover to grow linearly. Additionally, Trebi-Ollennu et al. (2001) describe the design and testing of a sun sensor on one of the FIDO rover platforms, reporting errors in rover heading of a few degrees. However, to economize, the MERs were launched without dedicated sun sensors and instead use the PANCAM stereo camera pair to search for and acquire images of the sun (Eisenman et al., 2002). This sun sensing technique is only used as a periodic heading or attitude update, not a direct component of any online navigation algorithm. This sun sensing procedure had only been used about 100 times as of January 2007 (Maimone et al., 2007).

More recently, Furgale et al. (2011a) presented an experimental study of sun sensing as a rover navigational aid. Navigational information is estimated using sun measurements, a local clock, quasi-analytical models of solar ephemeris, and, in some cases, a gravity vector measurement from an inclinometer. An estimate utilizing multiple measurements is determined by minimizing a scalar weighted cost function. Using these techniques, the absolute heading of the rover was able to be determined to within a few degrees. In this thesis, these techniques were used to produce periodic attitude updates for comparison to our technique, in which the sun sensor measurements are used directly in the VO pipeline.

In summary, while significant research has been focused on visual odometry and sun sensor aided navigation, the algorithm presented in this thesis is the first to incorporate sun sensor measurements directly into the visual odometry formulation. This novel approach provides considerable benefits for planetary rover exploration, as detailed in the following chapters.

Chapter 3

Mathematical formulation

In this chapter, we outline the mathematical formulation of our visual odometry solution with sun sensor and inclinometer measurements. We start by establishing notational and mathematical conventions and defining our key coordinate frames. We then outline the derivation of the error terms for the stereo camera, sun sensor, inclinometer, and prior. Finally, we discuss the bundle adjustment formulation that allows us to incorporate sun sensor and inclinometer measurements into the estimation solution as they are acquired. This approach enables us to constantly correct the orientation estimate of VO using absolute orientation information, preventing superlinear growth of error.

3.1 Preliminaries

For the sake of clarity, we will briefly explain the notational scheme used in this thesis. A summary of these conventions has also been provided on page vi for reference purposes. Vectors are represented by boldface lowercase characters, and matrices by boldface uppercase characters. The identity matrix is represented by $\mathbf{1}$, while a zero matrix is written as $\mathbf{0}$. An overbar denotes the nominal value of a quantity, a hat over top of an element indicates that it is an estimate of a true quantity, and a tilde signifies a measured value. The vector $\boldsymbol{\rho}_a^{d,e}$ represents a translation from point $\mathbf{e}$ to point $\mathbf{d}$, expressed in coordinate frame $\underrightarrow{\mathcal{F}}_a$. The rotation matrix $\mathbf{C}_{a,b}$ rotates vectors expressed in $\underrightarrow{\mathcal{F}}_b$ into $\underrightarrow{\mathcal{F}}_a$. We use the following set of three standard rotation matrices, using the shorthand $c_\theta := \cos\theta$ and $s_\theta := \sin\theta$:

$$\mathbf{R}_x(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_\theta & s_\theta \\ 0 & -s_\theta & c_\theta \end{bmatrix}, \quad \mathbf{R}_y(\theta) = \begin{bmatrix} c_\theta & 0 & -s_\theta \\ 0 & 1 & 0 \\ s_\theta & 0 & c_\theta \end{bmatrix}, \quad \mathbf{R}_z(\theta) = \begin{bmatrix} c_\theta & s_\theta & 0 \\ -s_\theta & c_\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Given any 3×1 vector $\mathbf{r} = [r_1 \;\; r_2 \;\; r_3]^T$, we can also define the usual 3×3 skew-symmetric cross operator (Hughes, 1986):

$$\mathbf{r}^\times := \begin{bmatrix} 0 & -r_3 & r_2 \\ r_3 & 0 & -r_1 \\ -r_2 & r_1 & 0 \end{bmatrix}$$

We will now discuss notation for the linearization of rotations, which will be used extensively throughout this thesis. Let us define a rotation matrix $\mathbf{C}(\boldsymbol{\theta})$, where $\boldsymbol{\theta}$ is a 3×1 column of Euler angles that define the rotation. A perturbation in this rotation matrix can be written as follows:

$$\mathbf{C}\left(\bar{\boldsymbol{\theta}} + \delta\boldsymbol{\theta}\right) \approx \left(\mathbf{1} - \left(\mathbf{S}\left(\bar{\boldsymbol{\theta}}\right)\delta\boldsymbol{\theta}\right)^\times\right)\mathbf{C}\left(\bar{\boldsymbol{\theta}}\right) \tag{3.1}$$

where $\bar{\boldsymbol{\theta}}$ is the nominal value, $\delta\boldsymbol{\theta}$ is the perturbation, $\mathbf{S}\left(\bar{\boldsymbol{\theta}}\right)$ is the matrix relating rotation vectors to Euler angles evaluated at the operating point, and $\mathbf{C}\left(\bar{\boldsymbol{\theta}}\right)$ is the rotation matrix at the operating point. We note that this expression describes how a perturbation of a rotation matrix corresponds to a perturbation of Euler angles, $\bar{\boldsymbol{\theta}} + \delta\boldsymbol{\theta}$. Notationally, it is simpler to write this expression as

$$\mathbf{C}\left(\bar{\boldsymbol{\theta}} + \delta\boldsymbol{\theta}\right) = \left(\mathbf{1} - \delta\boldsymbol{\phi}^\times\right)\mathbf{C}\left(\bar{\boldsymbol{\theta}}\right), \tag{3.2}$$

where $\delta\boldsymbol{\phi} := \mathbf{S}\left(\bar{\boldsymbol{\theta}}\right)\delta\boldsymbol{\theta}$ is a 3×1 rotation vector (Hughes, 1986).

3.2 Important coordinate frames

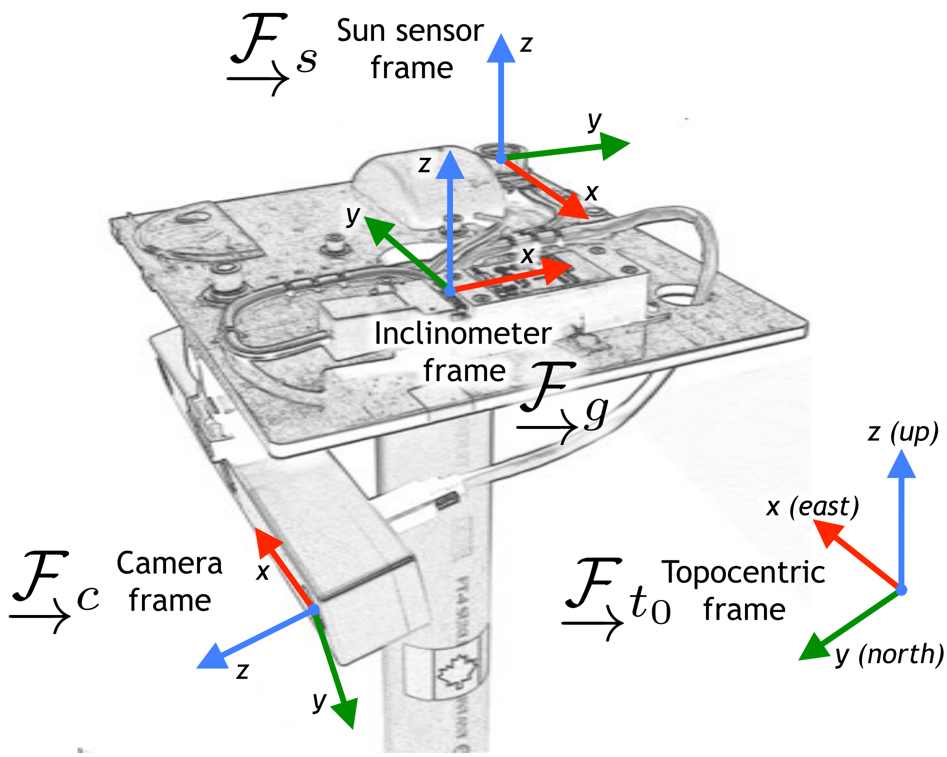


Figure 3.1: Illustration of the coordinate frames used in our formulation.

Our estimation framework relies on four main coordinate frames, with which we can describe all of our measurements and vehicle transformations. Figure 3.1 shows the sensor head used in our experiments, with each of the relevant coordinate frames defined.

The camera frame, $\underrightarrow{\mathcal{F}}_c$, is defined with origin at the left camera of the stereo apparatus. The x-axis is aligned with horizontal pixels, the y-axis with vertical pixels, and the z-axis is aligned with the optical axis. The sun sensor frame, $\underrightarrow{\mathcal{F}}_s$, is defined having a z-axis aligned with the outward normal of the sensor. For the inclinometer frame, $\underrightarrow{\mathcal{F}}_g$, the x- and y-axes of the frame are defined by the orthogonal sensing axes of the sensor. The locally defined topocentric frame, $\underrightarrow{\mathcal{F}}_t$, is such that the x-axis points in the eastward direction, the y-axis points north, and the z-axis is opposite to the local gravity vector.

With these definitions in hand, we can describe the pose of our rover. The estimate frame in our formulation, $\underrightarrow{\mathcal{F}}_{t_0}$, has the orientation of the topocentric frame and is located at the origin of our GPS unit at time $t = 0$. At each timestep k, we calculate the translation, $\boldsymbol{\rho}_{t_0}^{c_k,t_0}$, and rotation, $\mathbf{C}_{c_k,t_0}$, of the camera frame, $\underrightarrow{\mathcal{F}}_{c}$, relative to $\underrightarrow{\mathcal{F}}_{t_0}$, which can easily be transformed to a vehicle frame using calibration information.

3.3 Derivation of error terms

Given the stereo camera, sun sensor, and inclinometer measurements recorded at time k, our goal is to determine the maximum likelihood camera transformation at this timestep. Our method is to use a bundle adjustment approach that will estimate the states, $\{\mathbf{C}_{c_k,t_0}, \boldsymbol{\rho}_{t_0}^{c_k,t_0}\}$ and $\{\mathbf{C}_{c_{k-1},t_0}, \boldsymbol{\rho}_{t_0}^{c_{k-1},t_0}\}$, and the positions of the stereo camera landmarks, $\mathbf{p}_{t_0}^{j,t_0}$. Unlike the standard VO approach, which solves for a relative transformation from time $k-1$ to k, we solve for the states relative to $\underrightarrow{\mathcal{F}}_{t_0}$; this is because our new sun sensor and inclinometer measurement error terms require a current estimate of the vehicle orientation relative to the topocentric frame. In this technique, Gauss-Newton optimization is used to minimize an objective function composed of Mahalonobis distances proportional to the negative log likelihood of all the measurements. In order to build this objective function, we require error terms for each of the individual sensors, which we will derive in detail in this section.

3.3.1 Stereo camera model

We will now describe the observation models for our sensors, beginning with a stereo camera model. Note that we are not performing multi-frame visual odometry in this thesis, but instead only matching stereo keypoints frame to frame, as on the MERs. Accordingly, we begin with an initial stereo image pair at time $k-1$, and then a second at time k. Our goal is to determine the translation and rotation of the camera between the two image pairs by estimating the two states, $\{\mathbf{C}_{c_k,t_0}, \boldsymbol{\rho}_{t_0}^{c_k,t_0}\}$ and $\{\mathbf{C}_{c_{k-1},t_0}, \boldsymbol{\rho}_{t_0}^{c_{k-1},t_0}\}$. Our measurements are (u_l, v_l) and (u_r, v_r), the pixel locations of observed keypoints in the left and right rectified stereo images, respectively. The projection of landmark j, with three-dimensional location $\mathbf{p}_{c_k}^{j,c_k} = [x \;\; y \;\; z]^T$, into the image plane is described by our observation model for a stereo camera:

$$\mathbf{y}_k^j \;=\; \mathbf{h}\left(\mathbf{p}_{c_k}^{j,c_k}\right) = \begin{bmatrix} u_l \\ v_l \\ u_r \\ v_r \end{bmatrix} = \frac{1}{z}\begin{bmatrix} f_u(x+\frac{b}{2}) \\ f_v y \\ f_u(x-\frac{b}{2}) \\ f_v y \end{bmatrix} + \begin{bmatrix} c_u \\ c_v \\ c_u \\ c_v \end{bmatrix} + \delta\mathbf{y}_k^i \tag{3.3}$$

where f_u and f_v are the horizontal and vertical focal lengths in pixels, b is the camera baseline, and $\delta\mathbf{y}_k^j$ is the noise associated with each measurement, modelled as a zero-mean Gaussian density with covariance $\mathbf{T}_{y_k^j}$. Thus, we can define the error term for stereo measurements as follows:

$$\mathbf{e}_{y_k}^j := \mathbf{y}_k^j - \mathbf{h}\left(\mathbf{p}_{c_k}^{j,c_k}\right) \tag{3.4}$$

The three-dimensional location of the landmark relative to the camera frame at time k, $\mathbf{p}_{c_k}^{j,c_k}$, can be expressed as follows:

$$\mathbf{p}_{c_k}^{j,c_k} = \mathbf{C}_{c_k,t_0}\left(\mathbf{p}_{t_0}^{j,t_0} - \boldsymbol{\rho}_{t_0}^{c_k,t_0}\right) \tag{3.5}$$

In order to linearize the error term (3.4), we first perturb the landmark location (3.5) about its nominal value, as follows:

$$\begin{aligned} \mathbf{p}_{c_k}^{j,c_k} &= \bar{\mathbf{p}}_{c_k}^{j,c_k} + \delta\mathbf{p}_{c_k}^{j,c_k} \\ &\approx \left(\mathbf{1} - \delta\boldsymbol{\phi}_k^\times\right)\bar{\mathbf{C}}_{c_k,t_0}\left(\bar{\mathbf{p}}_{t_0}^{j,t_0} + \delta\mathbf{p}_{t_0}^{j,t_0} - \bar{\boldsymbol{\rho}}_{t_0}^{c_k,t_0} - \delta\boldsymbol{\rho}_{t_0}^{c_k,t_0}\right) \end{aligned}$$

Expanding and eliminating the products of small perturbation terms gives

$$
\begin{aligned}
\mathbf{p}_{c_k}^{j,c_k} \approx\ & \bar{\mathbf{C}}_{c_k,t_0}\left(\bar{\mathbf{p}}_{t_0}^{j,t_0} - \bar{\boldsymbol{\rho}}_{t_0}^{c_k,t_0}\right) + \bar{\mathbf{C}}_{c_k,t_0}\delta\mathbf{p}_{t_0}^{j,t_0} \\
& - \bar{\mathbf{C}}_{c_k,t_0}\delta\boldsymbol{\rho}_{t_0}^{c_k,t_0} + \delta\boldsymbol{\phi}_k^{\times}\bar{\mathbf{C}}_{c_k,t_0}\left(\bar{\boldsymbol{\rho}}_{t_0}^{c_k,t_0} - \bar{\mathbf{p}}_{t_0}^{j,t_0}\right).
\end{aligned}
$$

Using the identity $\mathbf{a}^{\times}\mathbf{b} \equiv -\mathbf{b}^{\times}\mathbf{a}$, we can rewrite the perturbed landmark position as

$$
\begin{aligned}
\mathbf{p}_{c_k}^{j,c_k} =\ & \bar{\mathbf{C}}_{c_k,t_0}\left(\bar{\mathbf{p}}_{t_0}^{j,t_0} - \bar{\boldsymbol{\rho}}_{t_0}^{c_k,t_0}\right) + \bar{\mathbf{C}}_{c_k,t_0}\delta\mathbf{p}_{t_0}^{j,t_0} \\
& - \bar{\mathbf{C}}_{c_k,t_0}\delta\boldsymbol{\rho}_{t_0}^{c_k,t_0} + \left(\bar{\mathbf{C}}_{c_k,t_0}\left(\bar{\mathbf{p}}_{t_0}^{j,t_0} - \bar{\boldsymbol{\rho}}_{t_0}^{c_k,t_0}\right)\right)^{\times}\delta\boldsymbol{\phi}_k.
\end{aligned}
$$

This equation can be written in matrix form as

$$
\mathbf{p}_{c_k}^{j,c_k} = \bar{\mathbf{p}}_{c_k}^{j,c_k} + \begin{bmatrix}\mathbf{D}_x & \mathbf{D}_p\end{bmatrix}\begin{bmatrix}\delta\mathbf{x} \\ \delta\mathbf{p}\end{bmatrix}, \tag{3.6}
$$

where

$$
\bar{\mathbf{p}}_{c_k}^{j,c_k} = \bar{\mathbf{C}}_{c_k,t_0}\left(\bar{\mathbf{p}}_{t_0}^{j,t_0} - \bar{\boldsymbol{\rho}}_{t_0}^{c_k,t_0}\right), \qquad \mathbf{D}_x = \begin{bmatrix}\mathbf{0} & \mathbf{0} & -\bar{\mathbf{C}}_{c_k,t_0} & \left(\bar{\mathbf{C}}_{c_k,t_0}\left(\bar{\mathbf{p}}_{t_0}^{j,t_0} - \bar{\boldsymbol{\rho}}_{t_0}^{c_k,t_0}\right)\right)^{\times}\end{bmatrix},
$$

$$
\mathbf{D}_p = \bar{\mathbf{C}}_{c_k,t_0}, \qquad \delta\mathbf{x} = \begin{bmatrix}\delta\boldsymbol{\rho}_{t_0}^{c_{k-1},t_0} \\ \delta\boldsymbol{\phi}_{k-1} \\ \delta\boldsymbol{\rho}_{t_0}^{c_k,t_0} \\ \delta\boldsymbol{\phi}_k\end{bmatrix}, \qquad \delta\mathbf{p} = \delta\mathbf{p}_{t_0}^{j,t_0}.
$$

Substituting (3.6) into (3.4), we obtain our linearized error term:

$$\begin{aligned}
\mathbf{e}_{y_k}^j &\approx \mathbf{y}_k^j - \mathbf{h}\left(\bar{\mathbf{p}}_{c_k}^{j,c_k} + \begin{bmatrix}\mathbf{D}_x & \mathbf{D}_p\end{bmatrix}\begin{bmatrix}\delta\mathbf{x}\\ \delta\mathbf{p}\end{bmatrix}\right) \\
&\approx \mathbf{y}_k^j - \mathbf{h}\left(\bar{\mathbf{p}}_{c_k}^{j,c_k}\right) - \left.\frac{\partial\mathbf{h}}{\partial\mathbf{p}}\right|_{\bar{\mathbf{p}}_{c_k}^{j,c_k}} \begin{bmatrix}\mathbf{D}_x & \mathbf{D}_p\end{bmatrix}\begin{bmatrix}\delta\mathbf{x}\\ \delta\mathbf{p}\end{bmatrix} \\
&= \mathbf{y}_k^j - \mathbf{h}\left(\bar{\mathbf{p}}_{c_k}^{j,c_k}\right) - \begin{bmatrix}\mathbf{A}_k^j & \mathbf{B}_k^j\end{bmatrix}\begin{bmatrix}\delta\mathbf{x}\\ \delta\mathbf{p}\end{bmatrix} \qquad (3.7)
\end{aligned}$$

where

$$\mathbf{A}_k^j = \mathbf{E}_k\mathbf{D}_x, \qquad \mathbf{B}_k^j = \mathbf{E}_k\mathbf{D}_p, \qquad \mathbf{E}_k = \left.\frac{\partial\mathbf{h}}{\partial\mathbf{p}}\right|_{\bar{\mathbf{p}}_{c_k}^{j,c_k}}.$$

Following the same logical progression, an analogous error term can be derived for the same landmark observed at time $k-1$, which will depend on the state perturbation terms, $\delta\boldsymbol{\rho}_{t_0}^{c_{k-1},t_0}$ and $\delta\boldsymbol{\phi}_{k-1}$:

$$\mathbf{e}_{y_{k-1}}^j = \mathbf{y}_{k-1}^j - \mathbf{h}\left(\bar{\mathbf{p}}_{c_{k-1}}^{j,c_{k-1}}\right) - \begin{bmatrix}\mathbf{A}_{k-1}^j & \mathbf{B}_{k-1}^j\end{bmatrix}\begin{bmatrix}\delta\mathbf{x}\\ \delta\mathbf{p}\end{bmatrix} \qquad (3.8)$$

where

$$\mathbf{A}_{k-1}^j = \mathbf{E}_{k-1}\begin{bmatrix}-\bar{\mathbf{C}}_{c_{k-1},t_0} & \left(\bar{\mathbf{C}}_{c_{k-1},t_0}\left(\bar{\mathbf{p}}_{t_0}^{j,t_0} - \bar{\boldsymbol{\rho}}_{t_0}^{c_{k-1},t_0}\right)\right)^{\times} & \mathbf{0} & \mathbf{0}\end{bmatrix},$$

$$\mathbf{B}_{k-1}^j = \mathbf{E}_{k-1}\bar{\mathbf{C}}_{c_{k-1},t_0}, \qquad \mathbf{E}_{k-1} = \left.\frac{\partial\mathbf{h}}{\partial\mathbf{p}}\right|_{\bar{\mathbf{p}}_{c_{k-1}}^{j,c_{k-1}}}.$$

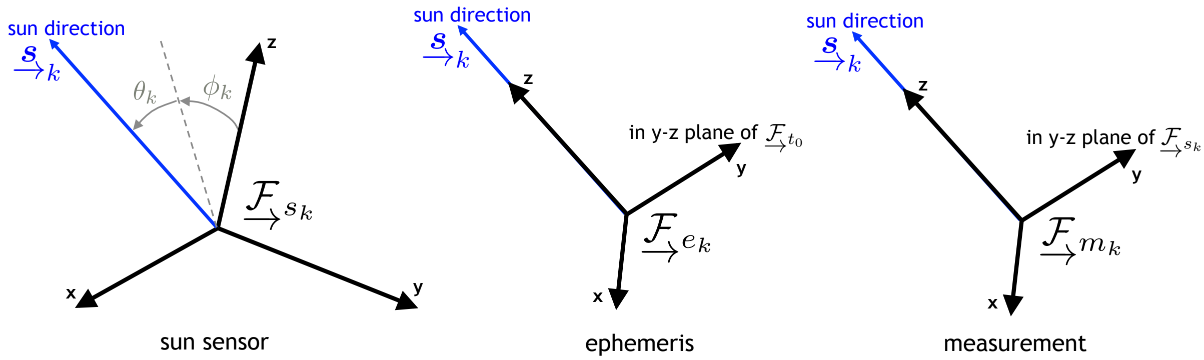


Figure 3.2: Definitions of the sun sensor measurement angles, ϕ_k and θ_k, as well as the ephemeris and measurement frames.

3.3.2 Sun sensor model

We will now outline the derivation of the sun sensor observation model, which closely resembles the one derived by Barfoot et al. (2010a). After image acquisition and some post-processing, the sun sensor determines a unit vector pointing from the sensor to the sun. This unit vector can be completely described relative to the sun sensor frame by the measurement $\mathbf{s}_k$, consisting of a rotation about the x-axis by angle ϕ_k and a rotation about the y-axis by θ_k. The definition of these angles is shown in Figure 3.2. In order to perform our analysis, we also define an ephemeris frame, $\underrightarrow{\mathcal{F}}_{e_k}$, with z-axis aligned with the sun direction, $\underrightarrow{s}_k$, and y-axis lying in the yz-plane of the topocentric frame, $\underrightarrow{\mathcal{F}}_{t_0}$. Additionally, we define a measurement frame, $\underrightarrow{\mathcal{F}}_{m_k}$, with z-axis aligned with the sun direction, $\underrightarrow{s}_k$, and y-axis lying in the yz-plane of the sun sensor frame, $\underrightarrow{\mathcal{F}}_{s_k}$.

Thus, we can write the set of Euler angles from our current measurement, $\tilde{\boldsymbol{\eta}}_k$, as follows:

$$\mathbf{s}_k := \tilde{\boldsymbol{\eta}}_k = \begin{bmatrix} \theta_k \\ \phi_k \end{bmatrix} \tag{3.9}$$

Based on these angle definitions, we can define the following Euler sequence for $\mathbf{C}_{s_k,m_k}$, our measurement:

$$\mathbf{C}_{s_k,m_k} = \mathbf{R}_x(\phi_k)\mathbf{R}_y(\theta_k)\mathbf{R}_z(0) \tag{3.10}$$

We can also define an Euler sequence for $\mathbf{C}_{e_k,m_k}$, which is a rotation about the z-axis (of either frame) through an unknown angle, ψ_k:

$$\mathbf{C}_{e_k,m_k} = \mathbf{R}_x(0)\mathbf{R}_y(0)\mathbf{R}_z(\psi_k) \tag{3.11}$$

With these definitions at hand, we can now go about deriving an error term for the sun sensor measurements. Noting that $\mathbf{C}_{s_k,m_k}$ contains the measurement information, we wish to build a predicted version of this, $\hat{\mathbf{C}}_{s_k,m_k}$. This predicted measurement will be based on our current attitude estimate relative to the topocentric frame, $\hat{\mathbf{C}}_{c_k,t_0}$, and the other interframe rotations:

$$\begin{aligned}
\hat{\mathbf{C}}_{s_k,m_k} &:= \mathbf{C}_{c,s}^T \hat{\mathbf{C}}_{c_k,t_0} \mathbf{C}_{t_0,e_k} \mathbf{C}_{e_k,m_k} \\
&= \mathbf{R}_x(\hat{\phi}_k)\mathbf{R}_y(\hat{\theta}_k)\mathbf{R}_z(\hat{\nu}_k)
\end{aligned}$$

Note that the rotation, $\mathbf{C}_{t_0,e_k}$, can be obtained from ephemeris, date and time of day, and an approximate knowledge of our current global position, and thus can be considered a known quantity. Also, the rotation, $\mathbf{C}_{c,s}$, is the rotation between the sun sensor and camera frames, and is assumed to be known from calibration. The angle $\hat{\nu}_k$ is non-zero because we are using $\hat{\mathbf{C}}_{c_k,t_0}$, not the true value, $\mathbf{C}_{c_k,t_0}$. Next, we can write the current timestep attitude estimate, $\hat{\mathbf{C}}_{c_k,t_0}$, as a multiplicative perturbation about some initial estimate, $\bar{\mathbf{C}}_{c_k,t_0}$, as follows:

$$\begin{aligned}
\hat{\mathbf{C}}_{s_k,m_k} &\approx \mathbf{C}_{c,s}^T \left(\mathbf{1} - \delta\boldsymbol{\phi}_k^\times\right) \bar{\mathbf{C}}_{c_k,t_0} \mathbf{C}_{t_0,e_k} \mathbf{C}_{e_k,m_k} \\
&= \mathbf{C}_{c,s}^T \bar{\mathbf{C}}_{c_k,t_0} \mathbf{C}_{t_0,e_k} \mathbf{C}_{e_k,m_k} - \mathbf{C}_{c,s}^T \delta\boldsymbol{\phi}_k^\times \bar{\mathbf{C}}_{c_k,t_0} \mathbf{C}_{t_0,e_k} \mathbf{C}_{e_k,m_k}
\end{aligned}$$

Next, we insert $\mathbf{C}_{c,s}\mathbf{C}_{c,s}^T$, which is equal to identity, into the expression:

$$\hat{\mathbf{C}}_{s_k,m_k} = \mathbf{C}_{c,s}^T \bar{\mathbf{C}}_{c_k,t_0} \mathbf{C}_{t_0,e_k} \mathbf{C}_{e_k,m_k} - \mathbf{C}_{c,s}^T \delta\boldsymbol{\phi}_k^\times \mathbf{C}_{c,s}\mathbf{C}_{c,s}^T \bar{\mathbf{C}}_{c_k,t_0} \mathbf{C}_{t_0,e_k} \mathbf{C}_{e_k,m_k}$$

Using the identity $(\mathbf{Cr})^{\times} \equiv \mathbf{C}\mathbf{r}^{\times}\mathbf{C}^{T}$, we can manipulate the expression to the following:

$$\hat{\mathbf{C}}_{s_k,m_k} = \left[\mathbf{1} - \left(\mathbf{C}_{c,s}^{T}\delta\boldsymbol{\phi}_k\right)^{\times}\right]\mathbf{C}_{c,s}^{T}\bar{\mathbf{C}}_{c_k,t_0}\mathbf{C}_{t_0,e_k}\mathbf{C}_{e_k,m_k} \tag{3.12}$$

The compound rotation, $\mathbf{C}_{c,s}^{T}\bar{\mathbf{C}}_{c_k,t_0}\mathbf{C}_{t_0,e_k}$, may be expressed as the following Euler sequence:

$$\mathbf{C}_{c,s}^{T}\bar{\mathbf{C}}_{c_k,t_0}\mathbf{C}_{t_0,e_k} = \mathbf{R}_x(\bar{\phi}_k)\mathbf{R}_y(\bar{\theta}_k)\mathbf{R}_z(\bar{\nu}_k)$$

Thus, we can write (3.12) as follows:

$$\begin{aligned}
\hat{\mathbf{C}}_{s_k,m_k} &= \left[\mathbf{1} - \left(\mathbf{C}_{c,s}^{T}\delta\boldsymbol{\phi}_k\right)^{\times}\right]\mathbf{R}_x(\bar{\phi}_k)\mathbf{R}_y(\bar{\theta}_k)\mathbf{R}_z(\bar{\nu}_k)\mathbf{R}_z(\psi_k) \\
&= \left[\mathbf{1} - \left(\mathbf{C}_{c,s}^{T}\delta\boldsymbol{\phi}_k\right)^{\times}\right]\mathbf{R}_x(\bar{\phi}_k)\mathbf{R}_y(\bar{\theta}_k)\mathbf{R}_z(\bar{\nu}_k + \psi_k) \\
&= \left[\mathbf{1} - \left(\mathbf{C}_{c,s}^{T}\delta\boldsymbol{\phi}_k\right)^{\times}\right]\bar{\mathbf{C}}_{s_k,m_k}
\end{aligned} \tag{3.13}$$

where $\bar{\mathbf{C}}_{s_k,m_k} = \mathbf{R}_x(\bar{\phi}_k)\mathbf{R}_y(\bar{\theta}_k)\mathbf{R}_z(\bar{\nu}_k + \psi_k)$. The angle ψ_k is unknown, but as we will see, we will not need it. The resulting expression for our predicted measurement is of the same linearized form as (3.2), with the $\delta\boldsymbol{\phi}_k$ from (3.2) taking the form $\mathbf{C}_{c,s}^{T}\delta\boldsymbol{\phi}_k$ in (3.13). Thus, the perturbation of the predicted measurement rotation matrix, as shown in (3.13), corresponds to a perturbation of the predicted measurement Euler angles, $\hat{\boldsymbol{\eta}}_k \approx \bar{\boldsymbol{\eta}}_k + \delta\boldsymbol{\eta}_k$. We can derive an expression for the perturbation $\delta\boldsymbol{\eta}_k$ using (3.1) and (3.2):

$$\begin{aligned}
\mathbf{C}_{c,s}^{T}\delta\boldsymbol{\phi}_k &= \mathbf{S}(\bar{\boldsymbol{\eta}}_k)\ \delta\boldsymbol{\eta}_k \\
\Rightarrow \delta\boldsymbol{\eta}_k &= \mathbf{S}^{-1}(\bar{\boldsymbol{\eta}}_k)\ \mathbf{C}_{c,s}^{T}\delta\boldsymbol{\phi}_k
\end{aligned} \tag{3.14}$$

Thus, the expression for the perturbed predicted measurement in Euler angles is as follows:

$$\hat{\boldsymbol{\eta}}_k = \bar{\boldsymbol{\eta}}_k + \mathbf{S}^{-1}(\bar{\boldsymbol{\eta}}_k)\ \mathbf{C}_{c,s}^{T}\delta\boldsymbol{\phi}_k \tag{3.15}$$

where

$$\hat{\boldsymbol{\eta}}_k := \begin{bmatrix} \hat{\nu}_k \\ \hat{\theta}_k \\ \hat{\phi}_k \end{bmatrix}, \quad \bar{\boldsymbol{\eta}}_k := \begin{bmatrix} \bar{\nu}_k + \psi_k \\ \bar{\theta}_k \\ \bar{\phi}_k \end{bmatrix}, \quad \mathbf{S}(\bar{\boldsymbol{\eta}}_k) := \begin{bmatrix} \mathbf{R}_x(\bar{\phi}_k)\mathbf{R}_y(\bar{\theta}_k)\mathbf{1}_3 & \mathbf{R}_x(\bar{\phi}_k)\mathbf{1}_2 & \mathbf{1}_1 \end{bmatrix}.$$

Note that the unknown angles $\bar{\nu}_k$ and ψ_k are not used in the calculation of $\mathbf{S}(\bar{\boldsymbol{\eta}}_k)$, due to our selection of Euler sequence. Additionally, our selection of Euler sequence has its singularity at $\theta_k = \pi/2$, which is out of the field of view of our sensor. We note, however, that our Euler angle terms consist of 3 unique angles, while our measurements consist of only 2 angles. To put our error expression in the same terms as our measurements, we utilize a projection matrix, $\mathbf{P}$, of the following form:

$$\mathbf{P} := \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Multiplying (3.15) by this projection matrix gives us the predicted sun sensor measurement in 2 Euler angles. Note that the application of the projection matrix makes the values of the unknown angles, $\bar{\nu}_k$ and ψ_k irrelevant. We can now write our final linearized expression for the sun sensor measurement error:

$$\begin{aligned} \mathbf{e}_{s_k} &= (\mathbf{s}_k - \hat{\boldsymbol{\eta}}_k) \\ &= \mathbf{s}_k - \mathbf{P}\left(\bar{\boldsymbol{\eta}}_k - \mathbf{S}^{-1}(\bar{\boldsymbol{\eta}}_k)\ \mathbf{C}_{c,s}^T \delta\boldsymbol{\phi}_k\right) \end{aligned} \tag{3.16}$$

We can reexpress the error term in matrix form as follows:

$$\mathbf{e}_{s_k} = \mathbf{s}_k - \mathbf{P}\bar{\boldsymbol{\eta}}_k + \mathbf{U}\,\delta\mathbf{x} \tag{3.17}$$

where

$$\mathbf{U} = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{P}\,\mathbf{S}^{-1}(\bar{\boldsymbol{\eta}}_k)\ \mathbf{C}_{c,s}^T \end{bmatrix}.$$

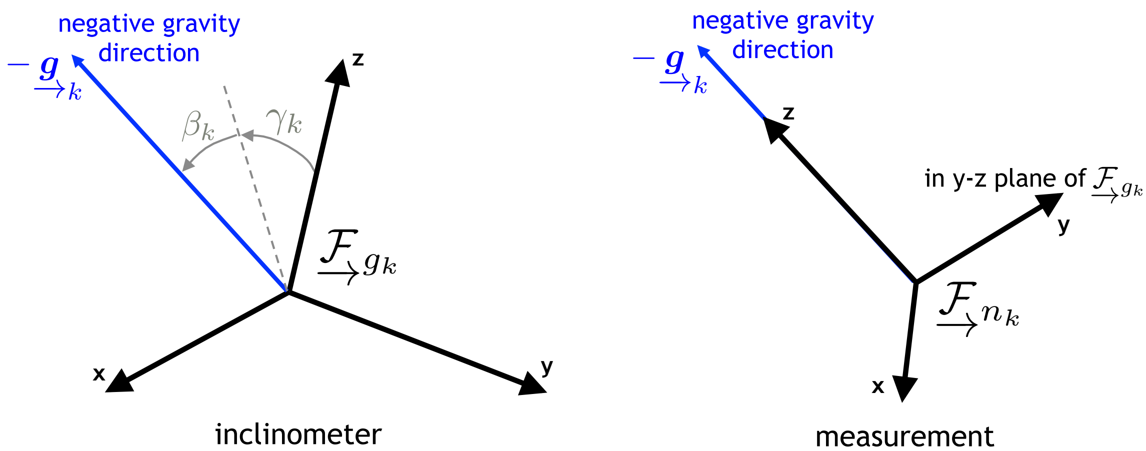


Figure 3.3: Definitions of the inclinometer measurement angles, β_k and γ_k, as well as the measurement frame.

3.3.3 Inclinometer model

The process of deriving the inclinometer model is essentially the same as for the sun sensor, so we present a very brief summary of the main concepts here. The inclinometer is measuring the pitch angle, β_k, and roll angle, γ_k, of the inclinometer with respect to the topocentric frame. The definition of these angles is shown in Figure 3.3, where we have shown the angles measured relative to the negative gravity vector, which is equivalent to the negative z-axis of the topocentric frame. We also define a measurement frame, $\underset{\rightarrow}{\mathcal{F}}_{n_k}$, with z-axis aligned with the negative gravity direction, $\underset{\rightarrow}{\boldsymbol{g}}_k$, and y-axis lying in the yz-plane of the inclinometer frame, $\underset{\rightarrow}{\mathcal{F}}_{g_k}$. Thus, we can write the set of Euler angles from our current measurement, $\tilde{\boldsymbol{\eta}}_k$, as follows:

$$\mathbf{g}_k := \tilde{\boldsymbol{\eta}}_k = \begin{bmatrix} \beta_k \\ \gamma_k \end{bmatrix} \tag{3.18}$$

Based on these angle definitions, we can define the following Euler sequence for $\mathbf{C}_{g_k,n_k}$, our measurement:

$$\mathbf{C}_{g_k,n_k} = \mathbf{R}_x(\gamma_k)\mathbf{R}_y(\beta_k)\mathbf{R}_z(0)$$

We can also define a Euler sequence for $\mathbf{C}_{t_0,n_k}$, which is a rotation about the z-axis (of either frame) through an unknown angle, ψ_k:

$$\mathbf{C}_{t_0,n_k} = \mathbf{R}_x(0)\mathbf{R}_y(0)\mathbf{R}_z(\psi_k)$$

Noting that $\mathbf{C}_{g_k,n_k}$ contains the measurement information, we can build a predicted version of this, $\hat{\mathbf{C}}_{g_k,n_k}$. This predicted measurement will be based on our current attitude estimate relative to the topocentric frame, $\hat{\mathbf{C}}_{c_k,t_0}$, and the other interframe rotations:

$$\begin{aligned} \hat{\mathbf{C}}_{g_k,n_k} &:= \mathbf{C}_{c,g}^T \hat{\mathbf{C}}_{c_k,t_0} \mathbf{C}_{t_0,n_k} \\ &= \mathbf{R}_x(\hat{\gamma}_k)\mathbf{R}_y(\hat{\beta}_k)\mathbf{R}_z(\hat{\nu}_k) \end{aligned}$$

Following the same procedure as outlined in the sun sensor model subsection, the linearized inclinometer error term can be determined from the above equations as follows:

$$\mathbf{e}_{g_k} = \mathbf{g}_k - \mathbf{P}\left(\bar{\boldsymbol{\eta}}_k - \mathbf{S}^{-1}(\bar{\boldsymbol{\eta}}_k)\ \mathbf{C}_{c,g}^T \delta\boldsymbol{\phi}_k\right) \tag{3.19}$$

We can reexpress the error term in matrix form as follows:

$$\mathbf{e}_{g_k} = \mathbf{g}_k - \mathbf{P}\bar{\boldsymbol{\eta}}_k + \mathbf{G}\,\delta\mathbf{x} \tag{3.20}$$

where

$$\mathbf{G} = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{P}\,\mathbf{S}^{-1}(\bar{\boldsymbol{\eta}}_k)\ \mathbf{C}_{c,g}^T \end{bmatrix}. \tag{3.21}$$

We note that (3.17) and (3.20) are of the same form, since both the sun and gravity directions are vector measurements described by a pair of rotations about the x- and y-axes.

3.3.4 Prior Term

As previously mentioned, because our sun sensor and inclinometer measurement error terms require a current estimate of the vehicle orientation relative to the topocentric frame, we solve for the states relative to $\underrightarrow{\mathcal{F}}_{t_0}$. To constrain this solution in space, we include a prior on the state variables based on the estimate up to the current time; this bundle adjustment implementation is functionally similar to the iterated extended Kalman filter. We start by taking our estimate of the state resulting from the bundle adjustment solution at the previous timestep, and reframing it as a prior incorporating the information from all our measurements up to the time $k-1$ (at the end of Section 3.4, we will show how this prior is determined from the estimated quantities at a given timestep). We denote this term as $\hat{\mathbf{x}} = \{\hat{\mathbf{C}}_{c_{k-1},t_0}, \hat{\boldsymbol{\rho}}_{t_0}^{c_{k-1},t_0}\}$, with 6×6 covariance matrix $\mathbf{T}_{k-1}$. With this prior defined, we go about deriving the error terms for the translation and rotation elements of the vehicle state.

For the translational terms, we start by defining the error between the prior and the true state as $\delta\hat{\boldsymbol{\rho}}_{t_0}^{c_{k-1},t_0}$, which we assume to be a small perturbation. The covariance of this error, $E\left[\delta\hat{\boldsymbol{\rho}}_{t_0}^{c_{k-1},t_0} \;\; \delta\hat{\boldsymbol{\rho}}_{t_0}^{c_{k-1},t_0\,T}\right]$, is described by the 3×3 upper left hand corner of $\mathbf{T}_{k-1}$. With these definitions at hand, we can write the following expression for the true state:

$$\boldsymbol{\rho}_{t_0}^{c_{k-1},t_0} \approx \hat{\boldsymbol{\rho}}_{t_0}^{c_{k-1},t_0} + \delta\hat{\boldsymbol{\rho}}_{t_0}^{c_{k-1},t_0} \tag{3.22}$$

Following a similar logic, we can also express the true translational state using our current state estimate plus some error perturbation:

$$\boldsymbol{\rho}_{t_0}^{c_{k-1},t_0} \approx \bar{\boldsymbol{\rho}}_{t_0}^{c_{k-1},t_0} + \delta\boldsymbol{\rho}_{t_0}^{c_{k-1},t_0} \tag{3.23}$$

By equating (3.22) and (3.23) and rearranging some terms, we can arrive at an error expression for the translational prior:

$$\mathbf{e}_{k-1_{\text{trans}}} := \delta\hat{\boldsymbol{\rho}}_{t_0}^{c_{k-1},t_0} = \bar{\boldsymbol{\rho}}_{t_0}^{c_{k-1},t_0} - \hat{\boldsymbol{\rho}}_{t_0}^{c_{k-1},t_0} + \delta\boldsymbol{\rho}_{t_0}^{c_{k-1},t_0} \tag{3.24}$$

The rotational prior error can be derived in a similar, albeit slightly more complicated, fashion. We start by defining the error between the prior and the true state as the rotation vector $\delta\boldsymbol{\psi}_{k-1}$, which we assume to be a small perturbation:

$$\delta\boldsymbol{\psi}_{k-1}^{\times} := \mathbf{1} - \mathbf{C}_{c_{k-1},t_0}\hat{\mathbf{C}}_{c_{k-1},t_0}^{T} \tag{3.25}$$

The covariance of this error, $E\left[\delta\boldsymbol{\psi}_{k-1}\;\delta\boldsymbol{\psi}_{k-1}{}^{T}\right]$, is described by the 3×3 lower right hand corner of $\mathbf{T}_{k-1}$. We can also express the true rotational state as a perturbation of our current state estimate:

$$\mathbf{C}_{c_{k-1},t_0} \approx \left(\mathbf{1} - \delta\boldsymbol{\phi}_{k-1}^{\times}\right)\bar{\mathbf{C}}_{c_{k-1},t_0} \tag{3.26}$$

We can combine these two expressions by substituting (3.26) into (3.25):

$$\delta\boldsymbol{\psi}_{k-1}^{\times} = \mathbf{1} - \left(\mathbf{1} - \delta\boldsymbol{\phi}_{k-1}^{\times}\right)\bar{\mathbf{C}}_{c_{k-1},t_0}\hat{\mathbf{C}}_{c_{k-1},t_0}^{T} \tag{3.27}$$

Noticing that the $\bar{\mathbf{C}}_{c_{k-1},t_0}\hat{\mathbf{C}}_{c_{k-1},t_0}^{T}$ term will be small, we can make the following approximation:

$$\bar{\mathbf{C}}_{c_{k-1},t_0}\hat{\mathbf{C}}_{c_{k-1},t_0}^{T} \approx \mathbf{1} - \delta\boldsymbol{\xi}_{k-1}^{\times} \tag{3.28}$$

where $\delta\boldsymbol{\xi}_{k-1}$ can be determined by computing the value of $\mathbf{1} - \bar{\mathbf{C}}_{c_{k-1},t_0}\hat{\mathbf{C}}_{c_{k-1},t_0}^{T} = \delta\boldsymbol{\xi}_{k-1}^{\times}$, and picking off the elements of $\delta\boldsymbol{\xi}_{k-1}$ from the resulting matrix. Substituting (3.28) into (3.27) and eliminating products of small perturbations yields

$$\delta\boldsymbol{\psi}_{k-1}^{\times} = \delta\boldsymbol{\phi}_{k-1}^{\times} + \delta\boldsymbol{\xi}_{k-1}^{\times}.$$

Applying the identity $\mathbf{a}^\times + \mathbf{b}^\times \equiv (\mathbf{a}+\mathbf{b})^\times$, we arrive at our final error expression for the rotational state prior:

$$\mathbf{e}_{k-1_{\mathrm{rot}}} := \delta\boldsymbol{\psi}_{k-1} = \delta\boldsymbol{\xi}_{k-1} + \delta\boldsymbol{\phi}_{k-1} \tag{3.29}$$

In order to use these error terms in the bundle adjustment solution, we also define the following matrices:

$$\mathbf{e}_{k-1} = \begin{bmatrix} \mathbf{e}_{k-1_{\mathrm{trans}}} \\ \mathbf{e}_{k-1_{\mathrm{rot}}} \end{bmatrix}, \qquad \mathbf{R} = \begin{bmatrix} \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} \end{bmatrix}$$

It is worth pointing out that our formulation only applies a prior to the vehicle state and not the landmarks. Because there is no prior introduced on the landmarks, we have not accounted for the fact that keypoints may be matched at multiple frames; in other words, we run the risk of overtrusting measurements. However, this is an inherent risk in any frame-to-frame VO system, where the additional complexity of carrying forward landmark estimates through time is undesirable. It is important to note that, while frame-to-frame bundle adjustment provides a good estimation framework for our application, the sun sensor and inclinometer error terms derived in this thesis could be incorporated into any number of solution methods. As such, this landmark uncertainty issue could be resolved by using the sun sensor and inclinometer measurements within a full SLAM solution. However, this approach falls beyond the scope of this thesis, as our goal was to improve accuracy without increasing computational cost significantly.

3.4 Bundle adjustment solution

Using our linearized error terms from above, we can construct our objective function, which we will minimize to obtain the maximum likelihood estimate for our new vehicle state:

$$\mathbf{J}(\mathbf{x},\mathbf{p}) := \frac{1}{2}\left(\mathbf{e}_{k-1}^T\mathbf{T}_{k-1}^{-1}\mathbf{e}_{k-1} + \mathbf{e}_g^T\mathbf{T}_g^{-1}\mathbf{e}_g + \mathbf{e}_s^T\mathbf{T}_s^{-1}\mathbf{e}_s + \mathbf{e}_y^T\mathbf{T}_y^{-1}\mathbf{e}_y\right) \tag{3.30}$$

where $\mathbf{T}_{k-1}^{-1}$, $\mathbf{T}_g^{-1}$, $\mathbf{T}_s^{-1}$, and $\mathbf{T}_y^{-1}$ represent the inverse covariance matrices for the previous state, inclinometer, sun sensor, and stereo feature measurements, respectively. We can minimize this objective function using Gauss-Newton optimization. An update step in the optimization process can be determined by augmenting the classic bundle adjustment update step (Brown, 1958) as follows:

$$\mathbf{H}^T\mathbf{T}^{-1}\mathbf{H}\begin{bmatrix}\delta\mathbf{x}\\ \delta\mathbf{p}\end{bmatrix} = -\mathbf{H}^T\mathbf{T}^{-1}\mathbf{e}(\bar{\mathbf{x}},\bar{\mathbf{p}}) \tag{3.31}$$

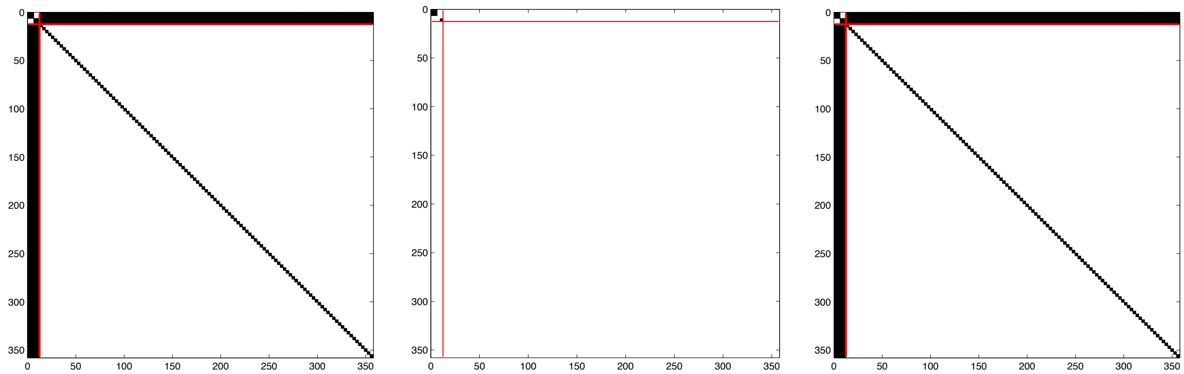
where

$$\mathbf{H} := \begin{bmatrix}\mathbf{R} & \mathbf{0}\\ \mathbf{G} & \mathbf{0}\\ \mathbf{U} & \mathbf{0}\\ \mathbf{A} & \mathbf{B}\end{bmatrix}, \qquad \mathbf{T}^{-1} := \operatorname{diag}\left\{\mathbf{T}_{k-1}^{-1}, \mathbf{T}_g^{-1}, \mathbf{T}_s^{-1}, \mathbf{T}_y^{-1}\right\},$$

and where $\mathbf{A}$ and $\mathbf{B}$ incorporate each of the Jacobians $\mathbf{A}_k^j$ and $\mathbf{B}_k^j$ for the keypoints observed at time k as well as the Jacobians $\mathbf{A}_{k-1}^j$ and $\mathbf{B}_{k-1}^j$ for the same keypoints observed at the time $k-1$. Constructing the left hand side, we have

$$\mathbf{H}^T\mathbf{T}^{-1}\mathbf{H} = \left[\begin{array}{c|c}\mathbf{R}^T\mathbf{T}_x^{-1}\mathbf{R} + \mathbf{G}^T\mathbf{T}_g^{-1}\mathbf{G} + \mathbf{U}^T\mathbf{T}_s^{-1}\mathbf{U} + \mathbf{A}^T\mathbf{T}_y^{-1}\mathbf{A} & \mathbf{A}^T\mathbf{T}_y^{-1}\mathbf{B}\\ \hline \mathbf{B}^T\mathbf{T}_y^{-1}\mathbf{A} & \mathbf{B}^T\mathbf{T}_y^{-1}\mathbf{B}\end{array}\right] \tag{3.32}$$

where the matrix is partitioned into the pose and landmark components.



(a) Contribution of stereo camera terms, as in (3.33).

(b) Contribution of prior, sun sensor, and inclinometer terms, as in (3.33).

(c) Summed matrix with all terms, as in (3.32).

Figure 3.4: The actual sparsity pattern of the $\mathbf{H}^T\mathbf{T}^{-1}\mathbf{H}$ matrix from one timestep of our data set, showing contributions from the stereo camera, prior, sun sensor, and inclinometer terms. Black indicates an occupied element, while white represents a zero. The red lines indicate the block partitioning shown in (3.32) and (3.33).

We can explicitly illustrate the contributions of the stereo camera, prior, sun sensor, and inclinometer terms by rewriting (3.32) as a sum:

$$\mathbf{H}^T\mathbf{T}^{-1}\mathbf{H} = \underbrace{\left[\begin{array}{c|c} \mathbf{A}^T\mathbf{T}_y^{-1}\mathbf{A} & \mathbf{A}^T\mathbf{T}_y^{-1}\mathbf{B} \\ \hline \mathbf{B}^T\mathbf{T}_y^{-1}\mathbf{A} & \mathbf{B}^T\mathbf{T}_y^{-1}\mathbf{B} \end{array}\right]}_{\text{stereo camera terms}} + \underbrace{\left[\begin{array}{c|c} \mathbf{R}^T\mathbf{T}_x^{-1}\mathbf{R} + \mathbf{G}^T\mathbf{T}_g^{-1}\mathbf{G} + \mathbf{U}^T\mathbf{T}_s^{-1}\mathbf{U} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} \end{array}\right]}_{\text{prior, sun sensor, and inclinometer terms}} \tag{3.33}$$

From (3.33), we can see that the inclusion of the additional measurements does not disturb the sparse structure of the bundle adjustment problem. The prior, sun sensor, and inclinometer terms are block diagonal (as shown in Figure 3.4(b)), and are added to the upper left corner of the stereo camera terms, which are also block diagonal (see Figure 3.4(a)). Thus, as shown in Figure 3.4(c), the sparse structure is preserved. This allows us to solve for the update step using computationally efficient sparse methods, as outlined in Appendix A.

We determine the state at time k, $\{\mathbf{C}_{c_k,t_0}, \boldsymbol{\rho}_{t_0}^{c_k,t_0}\}$, through an iterative sequence of update steps, as per Gauss-Newton (note that while we also solve for the state at time $k-1$, the new estimate is discarded, as it was necessary only to allow the addition of the prior term):

1. Using the current estimated values for $\{\mathbf{C}_{c_k,t_0}, \boldsymbol{\rho}_{t_0}^{c_k,t_0}\}$, $\{\mathbf{C}_{c_{k-1},t_0}, \boldsymbol{\rho}_{t_0}^{c_{k-1},t_0}\}$, and $\mathbf{p}_{t_0}^{j,t_0}$, compute an update step $\begin{bmatrix}\delta\mathbf{x}^T & \delta\mathbf{p}^T\end{bmatrix}^T$ by solving equation (3.31) using sparse bundle adjustment methods (see Appendix A).

2. Check for convergence. If converged, stop; otherwise continue to Step 3.

3. The state and feature position updates are then applied to $\{\mathbf{C}_{c_k,t_0}, \boldsymbol{\rho}_{t_0}^{c_k,t_0}\}$, $\{\mathbf{C}_{c_{k-1},t_0}, \boldsymbol{\rho}_{t_0}^{c_{k-1},t_0}\}$, and $\mathbf{p}_{t_0}^{j,t_0}$, respectively, according to

$$\mathbf{C}_{c_k,t_0} \leftarrow \boldsymbol{\Phi}_k \mathbf{C}_{c_k,t_0},$$

$$\boldsymbol{\rho}_{t_0}^{c_k,t_0} \leftarrow \boldsymbol{\rho}_{t_0}^{c_k,t_0} + \delta\boldsymbol{\rho}_{t_0}^{c_k,t_0},$$

$$\mathbf{p}_{t_0}^{j,t_0} \leftarrow \mathbf{p}_{t_0}^{j,t_0} + \delta\mathbf{p}_{t_0}^{j,t_0},$$

   where

$$\boldsymbol{\Phi}_k = \cos(\delta\phi_k)\,\mathbf{1} + (1-\cos(\delta\phi_k))\left(\frac{\delta\boldsymbol{\phi}_k}{\delta\phi_k}\right)\left(\frac{\delta\boldsymbol{\phi}_k}{\delta\phi_k}\right)^T - \sin(\delta\phi_k)\left(\frac{\delta\boldsymbol{\phi}_k}{\delta\phi_k}\right)^\times,$$

   and $\delta\phi_k := |\delta\boldsymbol{\phi}_k|$.

4. Return to Step 1.

Upon convergence, we obtain our maximum likelihood estimate for the current state at time k, $\{\bar{\mathbf{C}}_{c_k,t_0}, \bar{\boldsymbol{\rho}}_{t_0}^{c_k,t_0}\}$. Additionally, as shown in Appendix A, the 6×6 covariance matrix corresponding to this state, $\mathbf{T}_k$, is computed as part of the sparse bundle adjustment solution. With this information in hand, we can form the prior term that will be used at next timestep, $k+1$. The current state estimate, $\{\bar{\mathbf{C}}_{c_k,t_0}, \bar{\boldsymbol{\rho}}_{t_0}^{c_k,t_0}\}$, is redefined at the next timestep as the prior term, $\hat{\mathbf{x}} = \{\hat{\mathbf{C}}_{c_{k-1},t_0}, \hat{\boldsymbol{\rho}}_{t_0}^{c_{k-1},t_0}\}$. Similarly, the covariance matrix for the current state, $\mathbf{T}_k$, is used as the covariance matrix of the prior, $\mathbf{T}_{k-1}$.

This formulation allows us to incorporate sun sensor and inclinometer measurements directly into the visual odometry solution, as we acquire them. Note that we do not require both or any sun vector and gravity measurements at a given timestep; any measurements that are available can be individually included in the bundle adjustment solution. Thus, whenever possible, we are continuously correcting the camera orientation using absolute information, preventing the aggregation of errors over time. If there are no absolute orientation measurements at that timestep, the bundle adjustment will proceed as usual, solving for the camera transformation using only the stereo keypoints.

Chapter 4

Experimental Results

4.1 Description of field experiments

In order to test our visual odometry algorithm with sun sensor and inclinometer measurements, we utilized an extensive data set that was collected by Furgale et al. (2011b) in July 2008. A pushcart rover was taken on a 10 km traverse in a Mars analog environment on Devon Island, collecting stereo images, sun vectors, inclinometer gravity vectors, and GPS groundtruth for position. The details of this data set are presented in this section, including particulars of the traverse and the hardware configuration.

4.1.1 Scenario description

Data collection took place near the Haughton-Mars Project Research Station (HMPRS) (75°22' N latitude and 89°41' W longitude) on Devon Island in the Canadian High Arctic. The site is considered to be a strong analog for planetary environments (Lee et al., 1998; Barfoot et al., 2010b) due to its geological makeup and vegetation-free desert landscape, as seen in Figures 4.1 and 4.4. A pushcart rover platform traversed a 10 km loop through rugged canyons, sandy flats, and high-grade slopes over a period of 10 hours. To illustrate the path of the pushcart rover, Figure 4.2 shows the entire traverse loop plotted in Google Earth. The test was performed in July 2008; the Arctic sun remained above the horizon line for 24 hours a day.

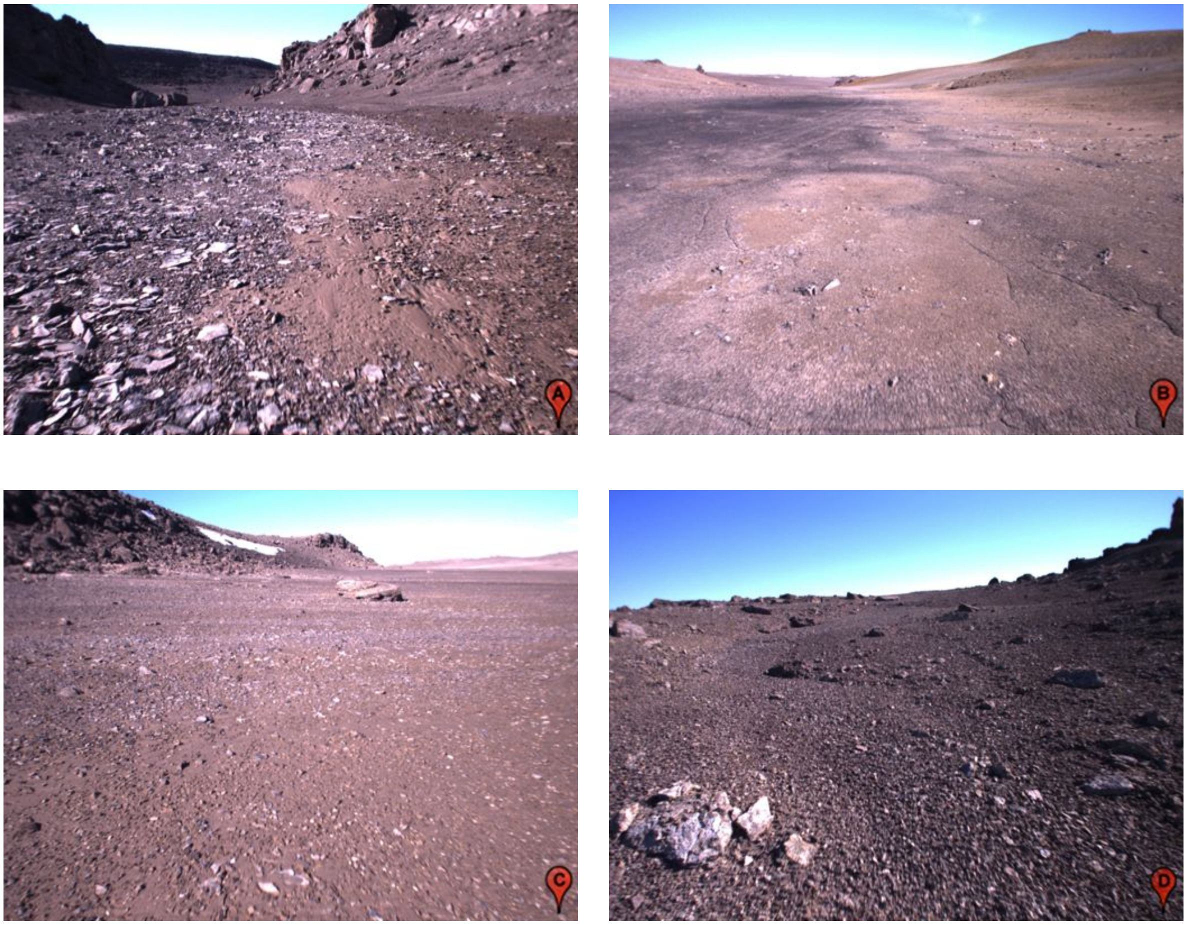


Figure 4.1: Typical images acquired by the stereo camera during the 10 km traverse of Devon Island. Note that for our experiments, we used 512×384 greyscale versions of the images. The letter labels are used to identify the image locations in Figure 4.2.

The 10 km loop has been partitioned into 23 sections of varying length, labelled with indices from 0 to 22. At the beginning of each of the individual sections, the rover platform remained stationary for a few minutes to collect large amounts of sun sensor and inclinometer data. These measurements have been used to provide periodic updates of the platform orientation for the full 10 km estimated traverse (Enright et al., 2009). It would also be possible to use these computed orientations to initialize the rover attitude of each VO path estimate, but any small angular error in the initial heading between our estimated path and our groundtruth path will lead to large errors over the course of hundreds of metres. To make the error reporting as accurate as possible, we have

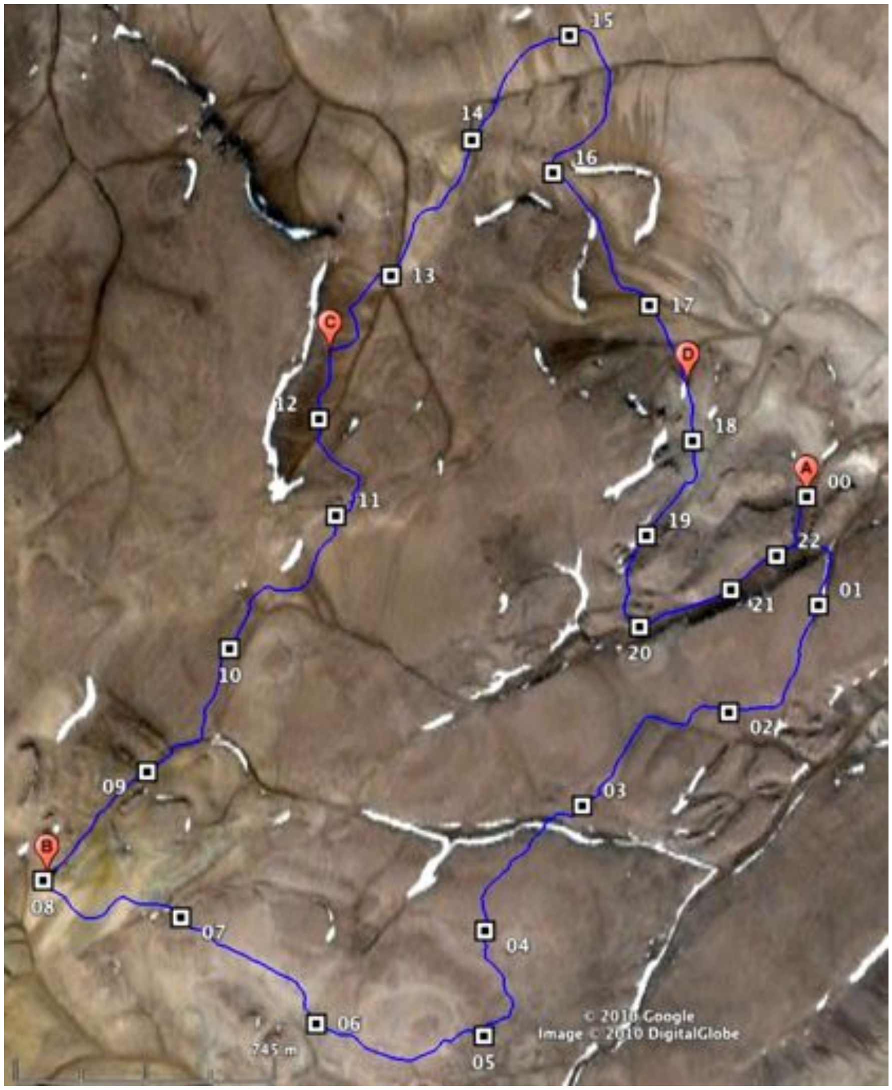
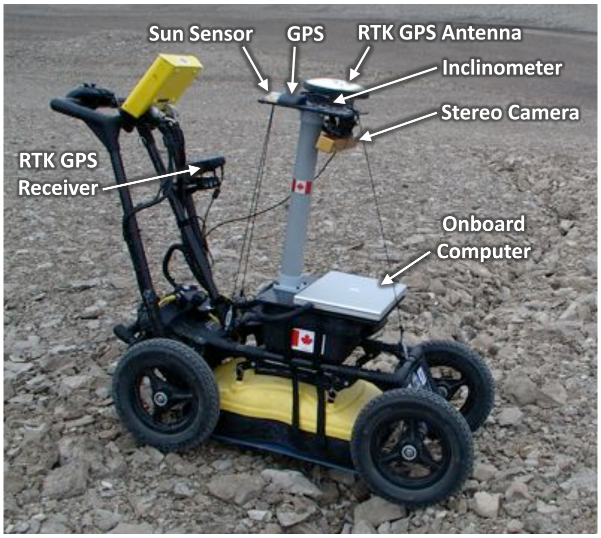
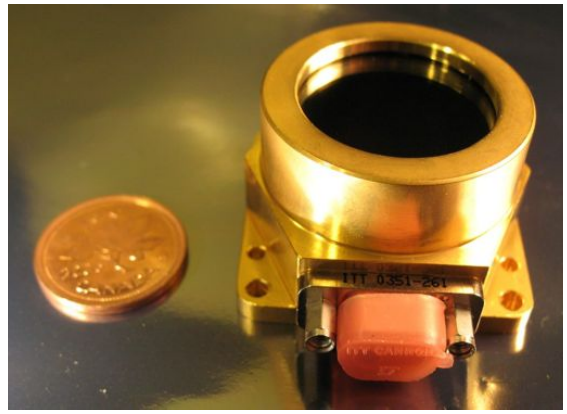


Figure 4.2: The 10 km traverse loop plotted in Google Earth. The starting points for the 23 individual sections are shown, as well as the locations of the images from Figure 4.1.

aligned the first 50 m of each estimated section traverse with GPS groundtruth, and then calculated the error on the remaining length of the traverse. This ensures that we are reporting the positional error accumulated by VO along the traverse path, and not reporting errors due to inaccurate initial heading. Positional groundtruth was provided by a pair of Magellan ProMark3 GPS units, which were used to produce post-processed differential GPS for the whole traverse. These measures of rover platform position and orientation allow us to confidently assess the accuracy of our motion estimates from VO.



(a) The pushcart rover platform at Devon Island.



(b) Sinclair Interplanetary SS-411 digital sun sensor.

Figure 4.3: The key hardware utilized in our field trials on Devon Island.

4.1.2 Hardware configuration

The data set was collected using a pushcart platform outfitted with a suite of rover engineering sensors, as seen in Figure 4.3(a). Since our visual odometry technique does not use any wheel odometry or rover telemetry data, the unactuated nature of the platform has no effect on our motion estimation. The stereo camera was a Point Grey Research Bumblebee XB3 with a baseline of 24 cm and a 70° field of view, mounted approximately 1 m above the surface pointing downward by approximately 20°. Over the course of the 10 km traverse, the stereo camera acquired 49410 images, which have been processed to 512×384 rectified greyscale images for these experiments. Color versions of some typical images from the data set are presented in Figure 4.1, further illustrating the Mars-like nature of the terrain. The inclinometer was a Honeywell HMR-3000, which weighs only 90 grams and uses a fluid tilt sensor to estimate the direction of gravity. Since the sensor directly outputs gravity vectors, the rover computer does not have to perform any expensive computation to produce these measurements.

Figure 4.4: The pushcart rover platform traversing on Devon Island.

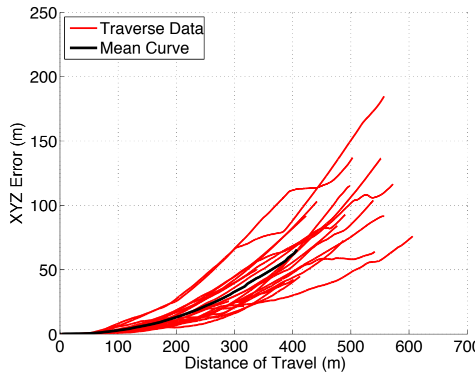
The sun sensor used in these experiments is a Sinclair Interplanetary SS-411 digital sun sensor, as shown in Figure 4.3(b). It is a low-power, low-mass device designed for use on microsatellites, weighing in at only 34 grams. A linear pixel array is used to capture an image, which is then processed by an integrated microcontroller to output floating point sun vector measurements in the sensor frame (Enright and Sinclair, 2007). Since the processing is done onboard by the sensor, producing sun vector measurements is of minimal computational cost to the rover computer. Additionally, the sensor monitors the quality of the detected images, rejecting poor quality images that can be caused by clouds or other factors. This produces intermittent gaps in the sun sensor data, which can subsequently affect our motion estimate. However, our VO framework is flexible enough to incorporate sun measurements when available and rely entirely on stereo vision when they are not.

4.2 Results

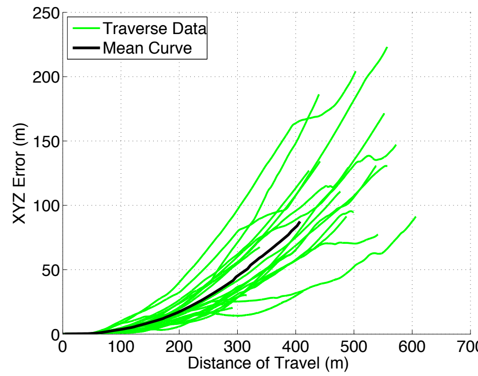
We will now present the experimental results produced by running our algorithm, as previously outlined in Chapter 3, offline on the 10 km loop data set from Devon Island. Motion estimates were calculated for each of the 23 individual sections of the traverse, allowing us to illustrate statistical trends on a large set of data. First, we evaluate the performance of three stereo feature detectors on the images from the Mars analog environment. Next, we examine the contributions of the sun sensor and inclinometer, and demonstrate the improved localization accuracy using both sensors on individual and concatenated sections. Finally, we present results illustrating how our algorithm can be used to decrease the number of images required by VO, thereby reducing its computational burden.

4.2.1 Evaluating feature detectors

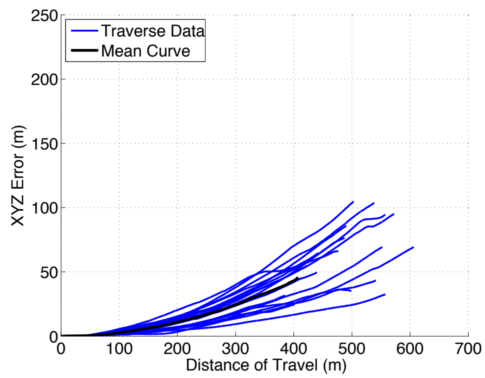
As detailed in Chapter 2, the first step in the visual odometry pipeline is to extract keypoints in the image. Section 2.1.1 introduced the Harris, FAST, and SURF detection algorithms, which have been commonly used in the literature. In order to evaluate which of these three detectors would perform best on our data, we ran our visual odometry algorithm (without sun sensor and inclinometer measurements) on all 23 individual sections of the dataset using each of the detectors. In order to make the comparison as fair as possible, we adjusted the parameters of the detectors such that each produced approximately the same number of detected keypoints for a given image, which were then passed to identical stereo matching and tracking algorithms. Also, for this comparison test, we did not utilize the keypoint scale values outputted by the SURF detector in any way, so as to maintain a level playing field.



(a) Harris.



(b) FAST.



(c) SURF.

Figure 4.5: XYZ error as a function of distance for all individual traverses, using the three feature detectors. Mean error curves have been included to aid comparison.

The results for each detector are shown in Figure 4.5, where the XYZ (i.e., Euclidean) error growth curves are shown for all 23 individual traverse sections. Additionally, to aid comparison between detectors, we have computed and plotted the mean error curves for the 23 traverse sections in Figure 4.6. In the interest of smoothness, the mean values were calculated in 5 m windows, and have only been computed up to a distance of travel where there are data available from 15 trials. As previously mentioned, for the purpose of accurate error reporting, we have aligned the first 50 m of each estimated path with groundtruth and the calculated the error on the remainder of the traverse. It is also important to note that some parameters in the algorithm required tuning, such as the disparity threshold for the stereo feature measurements. These values were tuned

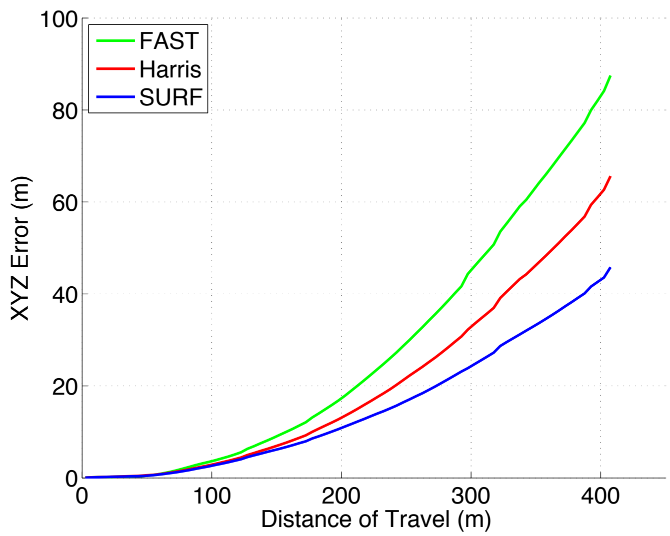


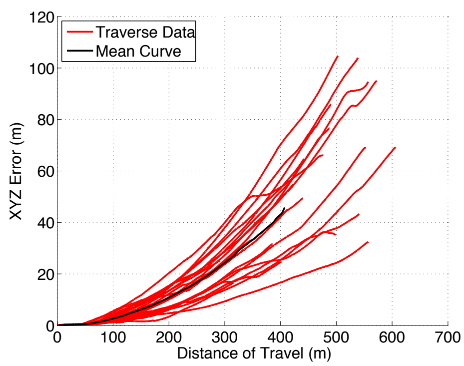
Figure 4.6: Mean XYZ error curves for all 23 traverse sections to aid comparison between the Harris, FAST, and SURF detectors.

to optimize performance on Section 0, and then those tuned parameters were retained for use with every other section. This procedure is similar to any field trial, in which parameters can be tuned based on a known set of data, and then tested experimentally in the field. Numerical results are presented in Table B.1 of Appendix B.

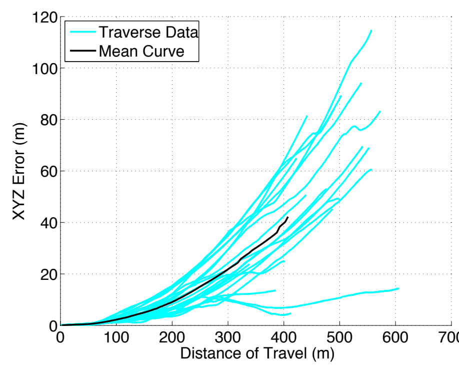
From Figure 4.5, we observe that some of the sections were quite challenging, producing large error compared to what has typically been reported in the literature. It is important to note that we are testing our technique on long distance traverses, over which the error of visual odometry is superlinearly increasing. Also, the challenging sections of the traverse often consisted of long stretches of flat terrain covered in small, pebble-like rocks. These flat, uniform landscapes are representative of common planetary environments, but they make rich feature detection difficult, producing the results we have observed. Also, unlike some of the more accurate techniques presented in the literature (Konolige et al., 2007; Mei et al., 2010), we are simply using frame-to-frame stereo matching, rather than a multiframe approach. Overall, the results indicate that the SURF detector is best suited to our challenging planetary analog terrain, as observed in Figure 4.6.

4.2.2 Using feature scale

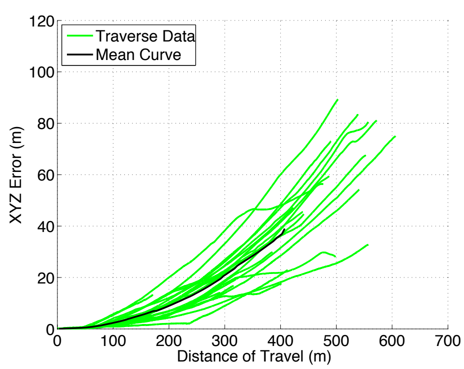
Given these results, we have selected the SURF detector to use in all our following visual odometry experiments incorporating a sun sensor and inclinometer. However, we first wish to examine whether we can further improve the accuracy of our motion estimates by utilizing the keypoint scale information to our advantage. The scale gives us some sense of our uncertainty in the location of the keypoint; specifically, a small keypoint scale indicates that the location of that feature in the image is known with little uncertainty, while a large keypoint scale implies great uncertainty.



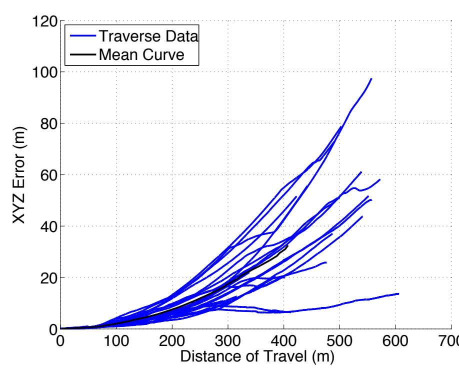
(a) Uniform matching, uniform covariance.



(b) Scale matching, uniform covariance.



(c) Uniform matching, scale covariance.



(d) Scale matching, scale covariance.

Figure 4.7: XYZ error as a function of distance for all individual traverses, using the varations on matching and covariance. Mean error curves have been included to aid comparison.

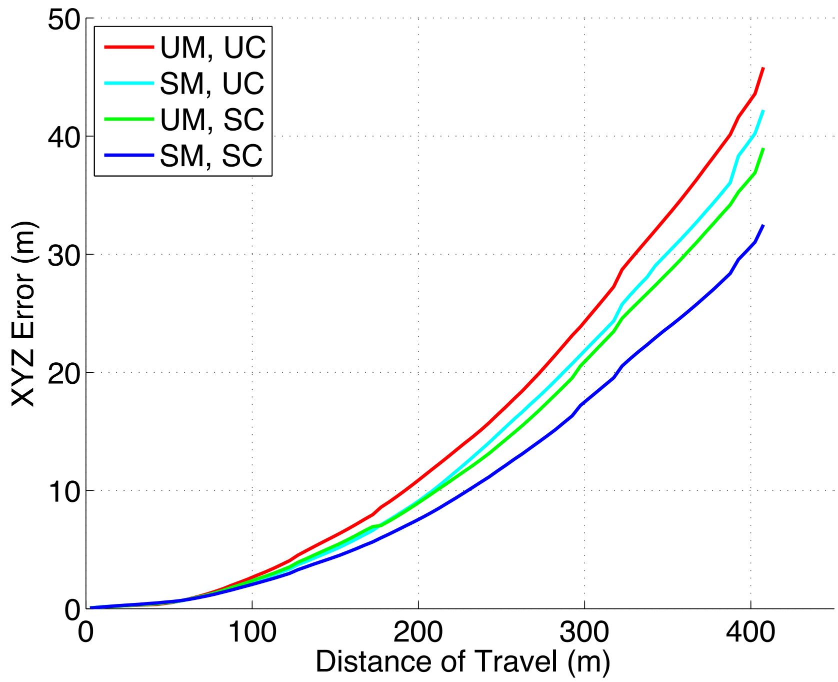


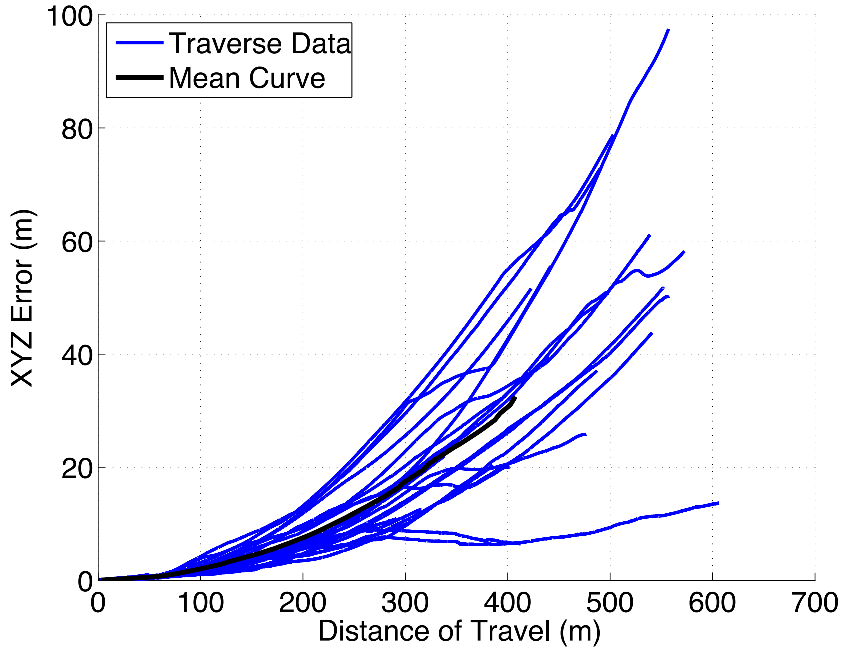
Figure 4.8: Mean XYZ error curves for all 23 traverse sections to aid comparison between uniform matching and uniform covariance (UM, UC), scale matching and uniform covariance (SM, UC), uniform matching and scale covariance (UM, SC), and scale matching and scale covariance (SM, SC), all using SURF features..

One way to use this information is to compute the covariance for each keypoint as a function of the keypoint scale, and employ each scale-specific covariance in the bundle adjustment solution. We can also use the scale information in the stereo matching step. Normally, we impose some threshold on how much a keypoint observed in left and right stereo images can violate the epipolar constraint, while still being considered a valid keypoint. Usually, this matching is performed with a uniform threshold over all keypoint scales, but a better approach would be to make this threshold a function of the keypoint uncertainty, or scale. We tested the effects of keypoint-scale based matching and covariance by computing visual odometry solutions using the same 23 traverse sections, producing the mean error curves presented in Figure 4.8. Numerical results are also presented in Table B.2 of Appendix B. We observe that employing the scale information in both of these techniques produces slightly more accurate motion estimates at a low additional cost. Thus, these modifications will be used in all following experiments.

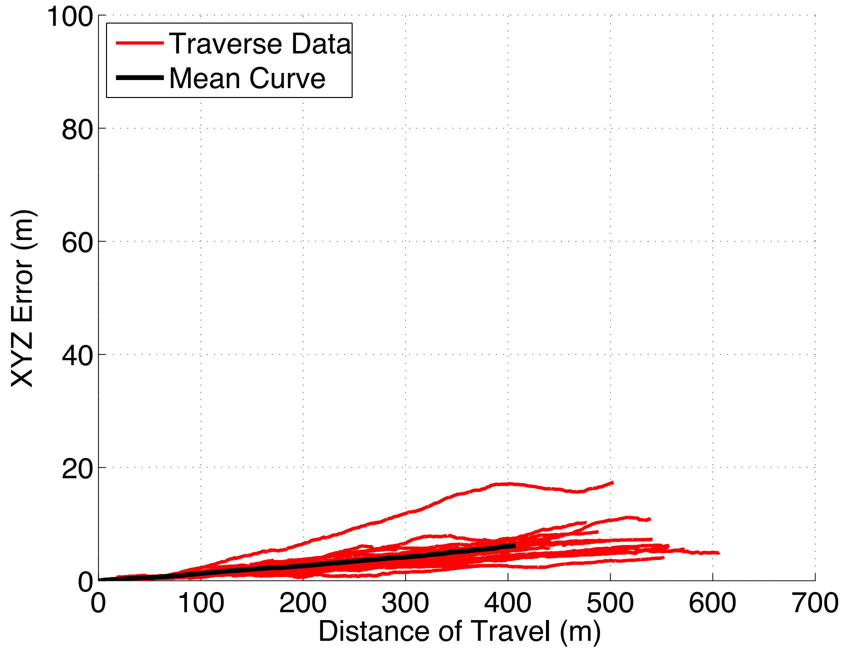
4.2.3 Evaluating the effects of the sun sensor and inclinometer

In this section, we present the motion estimates for each of the 23 sections produced by visual odometry incorporating sun sensor and inclinometer measurements. As determined in the above sections, these experiments were performed using the SURF feature detector with scale-based keypoint matching and covariance. We compare these results against the motion estimate produced using visual odometry only. Once again, the first 50 m segment of the estimated path is aligned with groundtruth, and all parameters are tuned on the first section and held constant for the remaining sections. The results are shown in Figure 4.9, where the XYZ error of the motion estimates has been plotted as a function of distance traversed. For ease of comparison, we have also included mean error curves computed in the same fashion as in Section 4.2.1 (mean values were calculated in 5 m windows, and have only been computed up to a distance of travel where there are data available from 15 trials). Numerical results are presented in Table B.3 of Appendix B, with the three-dimensional motion estimate error being expressed as a percentage of the total traversal distance.

In Figure 4.9(a), results are presented from the usual visual odometry algorithm, utilizing no additional measurements. As we have observed in the preceding sections, the error grows superlinearly with distance. Once again, this is because small orientation errors become amplified into large position errors over long distances. In contrast, Figure 4.9(b) shows that the addition of the sun sensor and inclinometer measurements consistently and dramatically limits the error growth on a large set of unique traverses. The constant orientation corrections from the sun sensor and inclinometer keep the platform attitude close to true, preventing the large amplification of errors and maintaining an approximately linear error growth.



(a) VO.



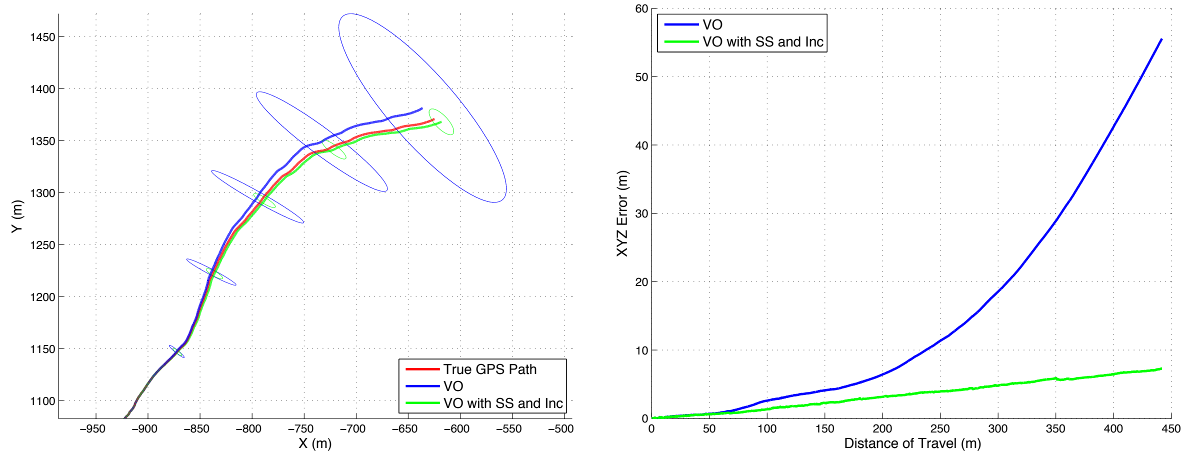
(b) VO with sun sensor and inclinometer.

Figure 4.9: XYZ error as a function of distance for all individual traverses, with mean error curves included.

The most striking thing about this improvement is that it comes at a very low cost. The rover needs not perform any computation to acquire the measurement vectors, and since there are many more stereo keypoints than sun or gravity vectors, the additional cost to bundle adjustment is almost negligible. A simple timing analysis using our MATLAB implementation showed that including the sun sensor and inclinometer measurements increased the computation time by approximately 0.3%. Given that our implementation is not currently optimized for speed, this is not meant to be taken as a formal timing test, but as an anecdotal demonstration that these additional measurements do not add a heavy computational burden to standard bundle adjustment.

In terms of the specific contributions of the sun sensor and inclinometer, the sun sensor will mainly provide information about the vehicle heading, and the inclinometer largely the pitch and roll. However, both sensors provide some measure of the full attitude of the rover. This is because, over the course of a traverse, the ground undulates and the sun moves in the sky, so the sensors will be measuring a continuous sequence of non-parallel vectors. In our experiments, we have found that the combination of both sensors greatly outperforms the use of either the sun sensor or the inclinometer alone. This is not only due to the fact that we are obtaining two distinct sets of vectors in nearly perpendicular directions, but we can also apply more frequent corrections.

To summarize, we have shown on a large set of individual traverses the statistical error improvement of VO with sun sensor and inclinometer measurements included. However, it is beneficial to focus in on a single typical example of these traverses, in order to glean some finer detail. For this thesis, we will focus on section 14, which demonstrates a typical error improvement. In order to visualize how our algorithm affects the motion estimate itself, we have presented the section 14 traverse and motion estimates in Figure 4.14(a). We observe that the heading error is greatly reduced, due to the corrections of the sun sensor, as well as the pitch error, due to the corrections of the inclinometer.



(a) Estimated motion path for Section 14, with 99.7% uncertainty ellipses shown.

(b) Error growth as a function of distance for Section 14. Note that the superlinear error growth of VO has been reduced to a more linear curve using sun sensor and inclinometer.

Figure 4.10: Estimated path results of our algorithm on Section 14 of the traverse, illustrating the accuracy improvement when using the sun sensor and inclinometer.

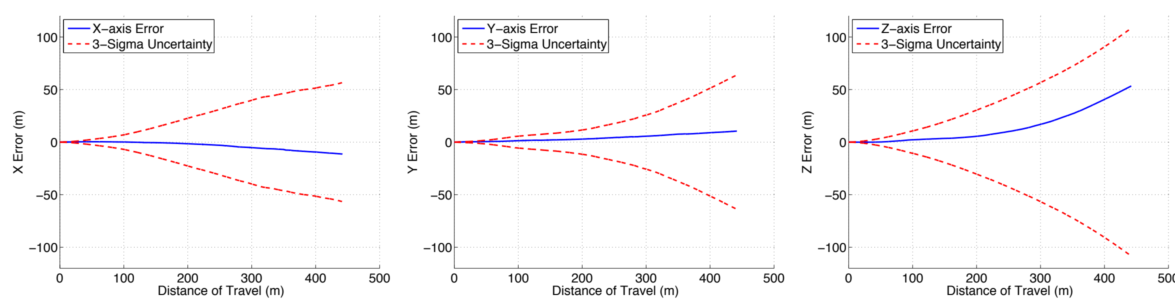


Figure 4.11: x- (left), y- (middle), and z- (right) axis error plots for the Section 14 traverse, estimated without sun sensor and inclinometer measurements.

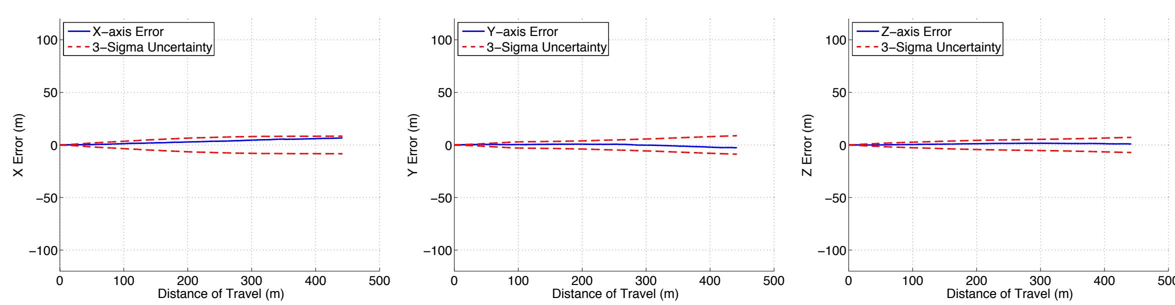


Figure 4.12: x- (left), y- (middle), and z- (right) axis error plots for the Section 14 traverse, estimated using sun sensor and inclinometer measurements.

Figure 4.14(a) also illustrates the growth of the 99.7% uncertainty ellipses over a number of timesteps from the traverse. We note that the incorporation of the sun sensor and inclinometer measurements significantly reduces the uncertainty in the robot's location at any given time. This result is shown in finer detail in Figures 4.11 and 4.12, illustrating the uncertainty envelopes for the Section 14 traverse. Once again, the significant accuracy improvements provided by the sun sensor and inclinometer are achieved with very low additional computational cost.

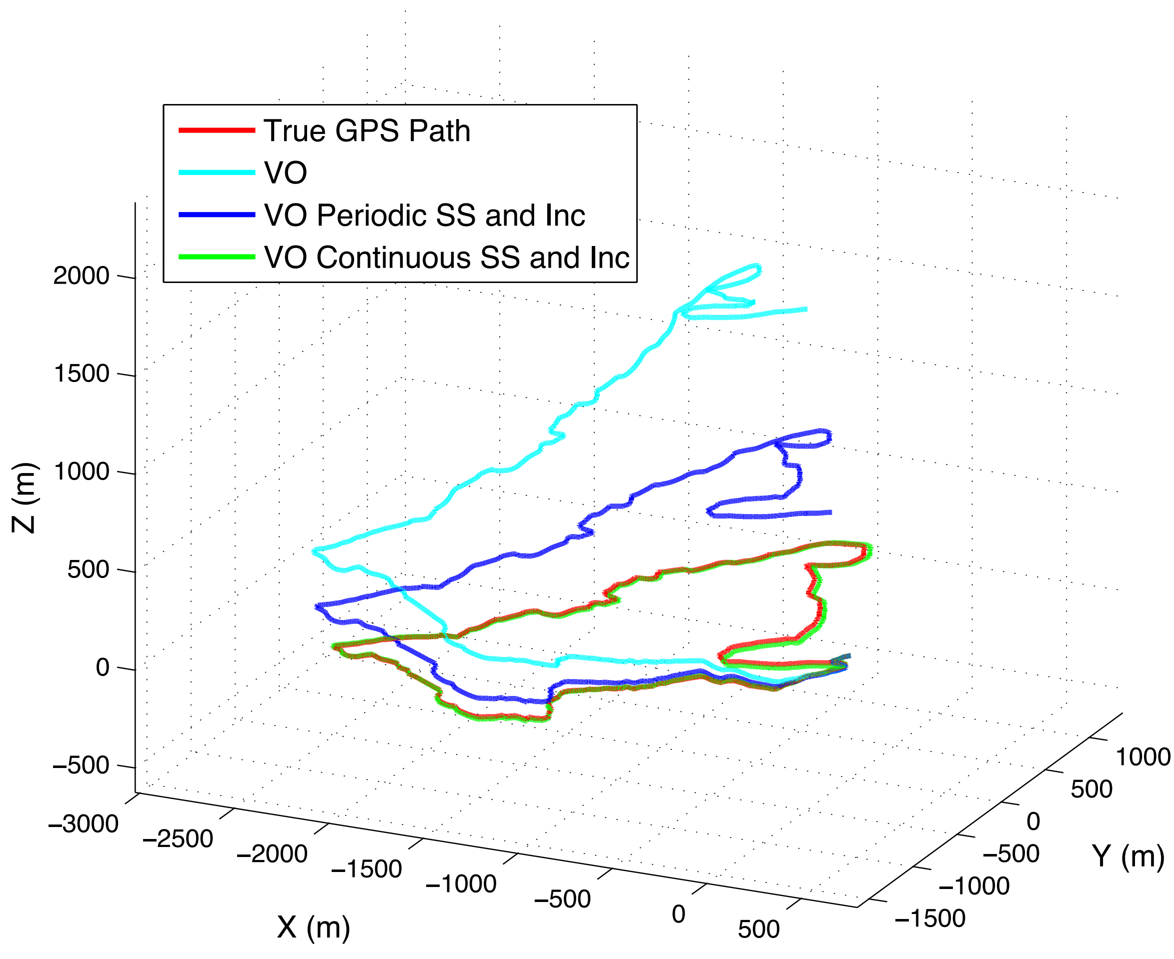
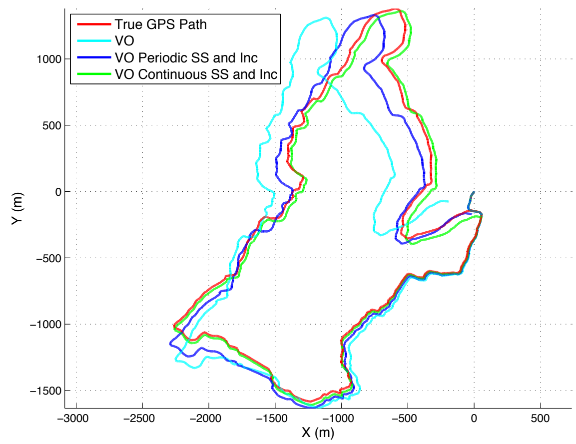
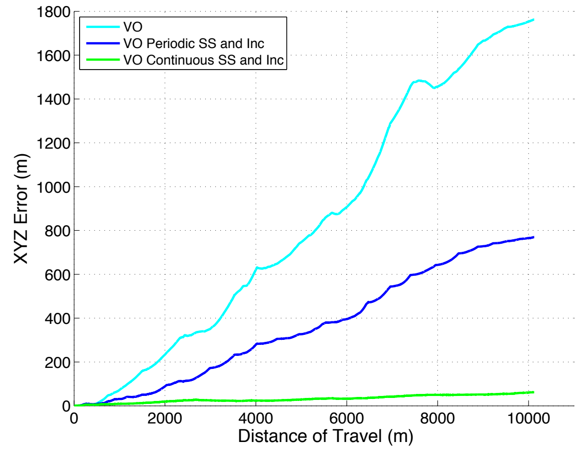


Figure 4.13: Three-dimensional view of motion estimate results for the full 10 km traverse.

As a final demonstration of the benefits of continuous sun sensor and inclinometer corrections, we have computed path estimates for the full 10 km traverse. The results are presented in Figures 4.13 and 4.14, with path estimates from three different versions of our



(a) x-y plane view of motion estimate results.



(b) Error growth as a function of distance.

Figure 4.14: Additional results from the full 10 km traverse, illustrating that the continuous orientation corrections from the sun sensor and inclinometer greatly limit error growth in the VO motion estimate.

algorithm. The first is straight visual odometry, with no additional measurements used at any point. The second variation is visual odometry with periodic orientation updates from the sun sensor and inclinometer at the start of each new section (approximately every 500 m), as previously demonstrated by Carle et al. (2010). These periodic updates are computed by allowing the vehicle to remain at rest for an extended period of time, in order to collect a large number of measurements. A batch solution method is then used to accurately compute the vehicle attitude, as described by Enright et al. (2009). This version of our algorithm is similar to the approach of the MERs, where sun measurements are periodically used to update the rover's orientation. The third version of our algorithm is visual odometry with continuous corrections from the sun sensor and inclinometer, as described in Chapter 3. Figure 4.13 illustrates that continuous correction of the vehicle's orientation using sun sensor and inclinometer measurements greatly restricts the error growth, producing an error of only 0.6% at the end of the 10 km traverse. We also observe that continuous corrections are significantly more accurate than only periodically updating the attitude, with very low additional computational cost.

In summary, over such a long traverse, we can significantly improve our visual odometry system with minimal additional computation, power, or mass, simply by including a sun sensor and inclinometer. To provide more detail about the terrain along the traverse, Figure 4.15 has also been included, showing the GPS altitude as a function of path length. The index and starting time of each individual traverse section has also been indicated.

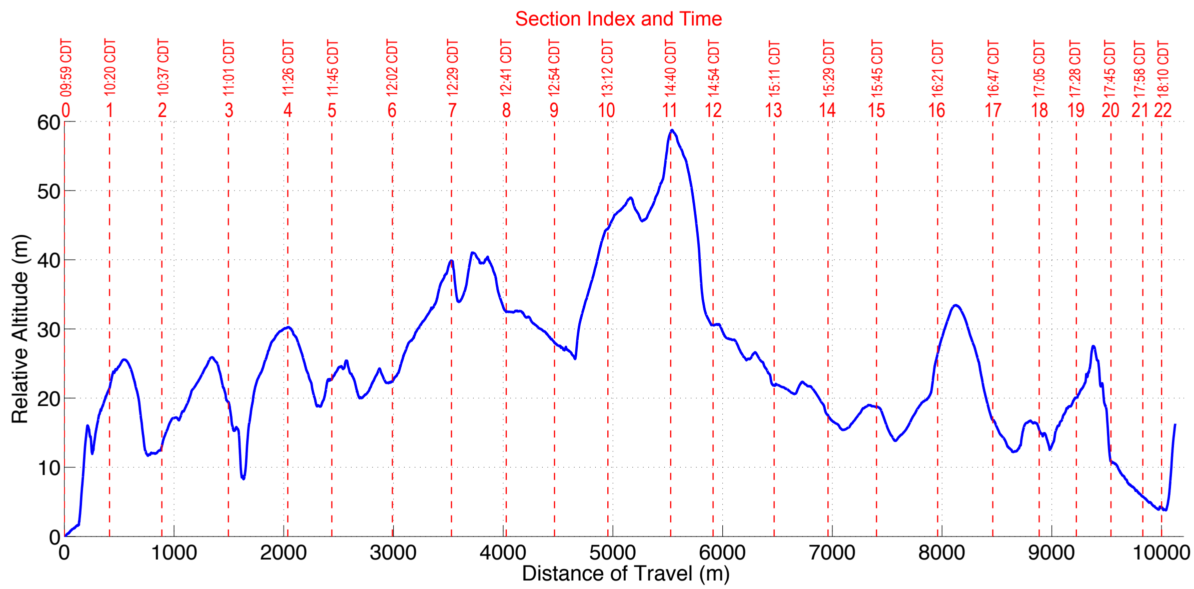


Figure 4.15: Altitude from GPS, relative to start of 10 km loop, as a function of distance. The individual traverse section indices have been indicated, as well as the Central Daylight Time (CDT) at the beginning of each traverse.

4.2.4 Using sun sensor and inclinometer measurements to reduce VO computation

In addition to the accuracy gains demonstrated in Section 4.2.3, we can also use the sun sensor and inclinometer measurements to lessen the computational requirements of visual odometry. As previously mentioned, the use of VO on the MERs has been severely limited by slow computation time. The hardware configuration of the rovers, including a space-qualified 20 MHz processor and slow-throughput camera bus, results in each VO update cycle taking up to 3 minutes to compute (Maimone et al., 2007). With a maximum spacing between images of 75 cm, the effective rover speed when using VO is approximately 10 m/h, an order of magnitude lower than the nominal rover speed (Maimone et al., 2007). Thus, in the interests of covering maximal ground, the use of VO has been limited to short drives (less than 15 m) involving specific terrain and tasks, despite its high positional accuracy. While the upcoming Mars Science Lab (MSL) rover employs a faster 200 MHz processor, its effective driving speed will still be limited by the VO update cycle time (Johnson et al., 2008; Matthies et al., 2007).

A simple approach to reduce this computational burden would be to acquire and process fewer stereo images per distance of travel. However, due to the increased spatial transformation of the camera between images, keypoint tracking becomes more difficult as the frame rate is decreased. With fewer keypoints, we would expect a less accurate visual odometry estimate. Our proposed algorithm compensates for this loss of accuracy by incorporating absolute orientation information from low-cost sun sensor and inclinometer measurements. The idea is to reduce the computational burden of visual ododmetry per distance of travel, while maintaining accuracy comparable to conventional VO.

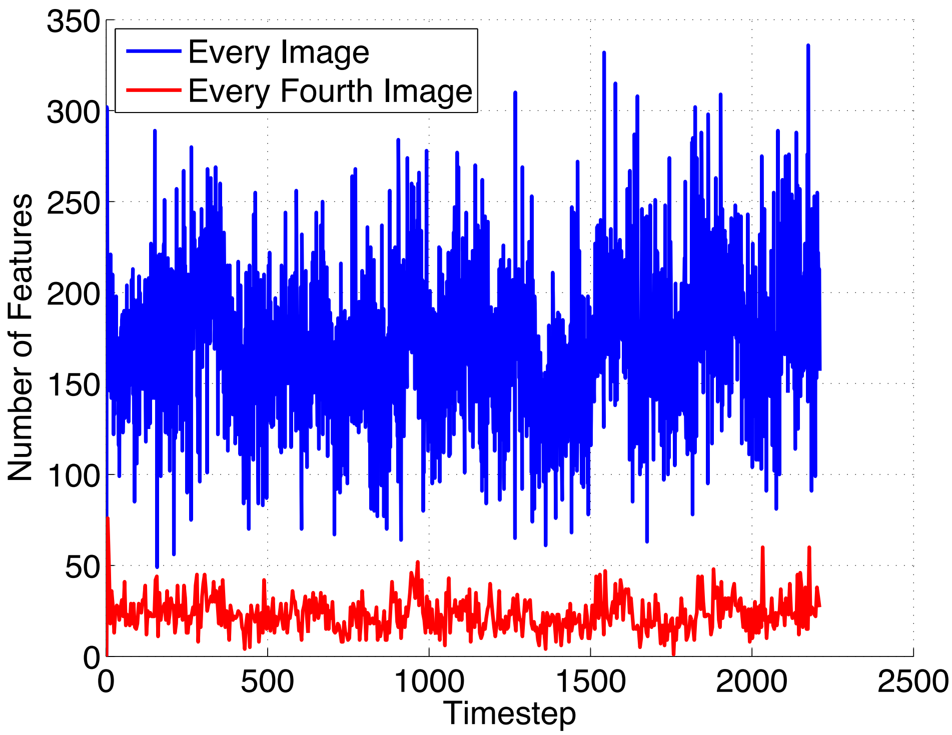


Figure 4.16: Number of tracked keypoints for traverse Section 14, illustrating the greatly reduced number of keypoints when processing every fourth frame.

As a demonstration of this concept, we gradually reduced the number of images in our Devon Island data set and computed corresponding VO estimates. Note that this also led to a reduced number of sun sensor and inclinometer readings, since these measurements must be associated with a stereo image acquired at approximately the same time. Perhaps counter-intuitively, we observed that initially reducing the frame rate actually produced less error in the motion estimate. This phenomena has been previously described by Howard (2008), who noted that higher visual odometry frame rates will integrate more noise into the motion estimate. While the accuracy did improve initially, further reducing the frame rate eventually resulted in sufficiently few tracked keypoints to produce a poor VO estimate. For our data set and algorithm, this was observed when every fourth stereo image was processed. Figure 4.16 illustrates the significantly reduced number of tracked keypoints at low frame rate for the representative Section 14 traverse. We then computed new motion estimates for all 23 traverse sections incorporating sun sensor and inclinometer measurements, producing the mean error curves observed in Figure 4.17. The absolute orientation corrections bring the low frame rate errors back into a comparable range with the high frame rate results, and, quite often, are even more accurate. This comparable error comes at a greatly reduced cost, since we are computing four times fewer VO updates, processing significantly less keypoints, and the additional cost of the sun sensor and inclinometer measurements is nearly negligible.

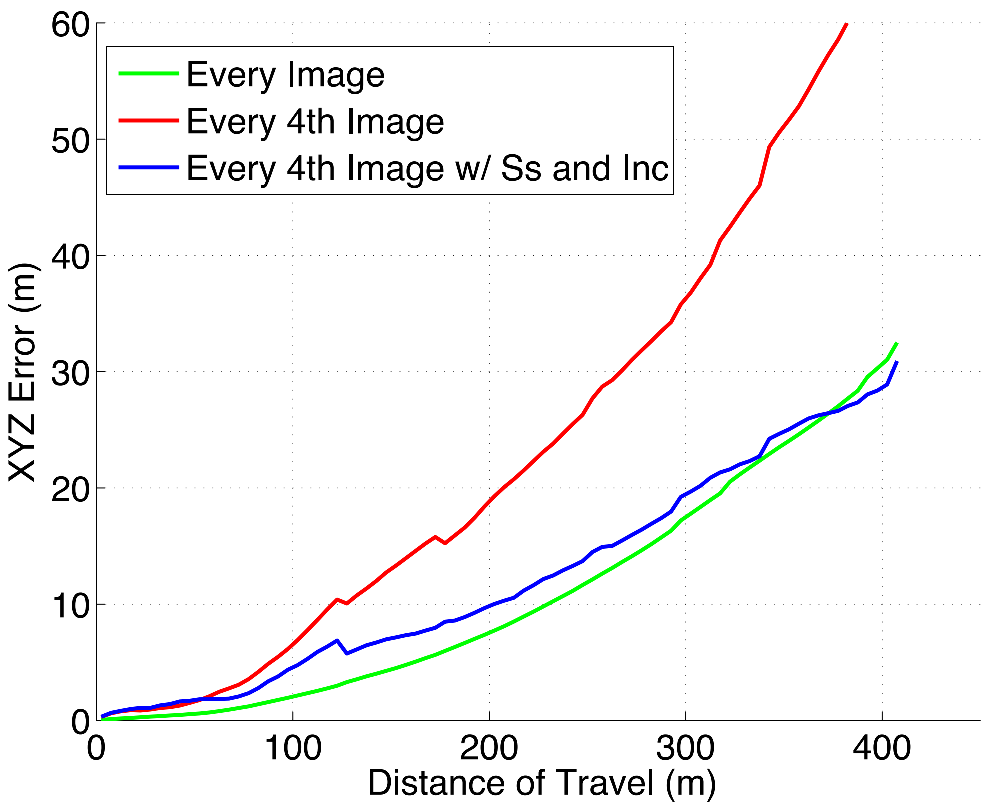


Figure 4.17: Mean XYZ error curves for varying frame rates, illustrating the average error for all 23 traverse sections. Note that the low frame rate result with sun sensor and inclinometer measurements included closely resembles the high frame rate result.

Chapter 5

Identifying Bias

Inspecting the full 10 km traverse from Chapter 4, it is interesting to note that the altitude estimate is responsible for most of the error; specifically, the motion estimate from regular visual odometry travels nearly 2 km up into the air. Upon further inspection of the individual traverse section estimates, we found that all but one ended up above the true path, as shown in Table B.4 of Appendix B. This result was particularly interesting because similar behaviour had been observed in the visual-teach-and-repeat experiments performed by Furgale and Barfoot (2010). Given the Gaussian noise model of our measurements, we would expect the motion estimates to be uniformly spread around the true paths, so it was a curious question: why does the estimate always seem to go up?

In order to tackle this problem, we needed to be able to isolate the variables; the upward bias could have been caused by anything from an incorrect camera calibration to a mathematical or coding error. In order to narrow things down, a simulation environment was created to generate artificial keypoint measurements which could be fed into our sparse bundle adjustment solver. The simulated data included all correspondences between keypoints, eliminating the need for any keypoint matching or tracking and allowing us to determine whether the bias was introduced in the measurements or in the computation of the pose estimate.

Figure 5.1: An illustration of how four keypoint measurements are used to model the oblique planetary terrain. Note how landmarks close to the camera appear at the bottom of the image, while far away landmarks appear at the top.

The measurements were generated as follows:

1. At time $k-1$, simulate an observed stereo keypoint measurement by selecting pixel locations within imaginary left and right images.
2. Project this measurement into a three-dimensional landmark using the camera model described in (3.3).
3. Move the the imaginary camera straight ahead forward, simulating rover motion.
4. Now at time k, re-observe the landmark as pixel measurements in the left and right cameras using the inverse of the camera model from (3.3).
5. Corrupt all pixel measurements with Gaussian noise.

At first, our approach was to generate a large number of random keypoint locations that were uniformly distributed throughout the scene. The VO estimates produced using these random keypoints had no discernible bias, indicating that the bundle adjustment implementation was not the root of the problem. So we tried an alternate keypoint configuration, using only four landmarks to very simply model the oblique terrain which a planetary rover typically observes. As seen in Figure 4.1, landmarks close to the camera appear in the bottom of the image, while far away landmarks are seen near the top. Our approach reproduces this oblique presentation of the landmarks to the camera using only four measurements, as shown in Figure 5.1.

Interestingly, using this oblique keypoint configuration produced VO results that were strongly biased in the upward direction. Figure 5.2 illustrates the results from 50 VO trials, each consisting of 1000 simulated images separated by a 0.5 m camera translation. As expected, the motion estimates produced using random landmarks are uniformly spread about the true path, due to the measurement noise. The estimates produced using the oblique landmarks, on the other hand, are spread about an upward trajectory, indicating that a bias is present.

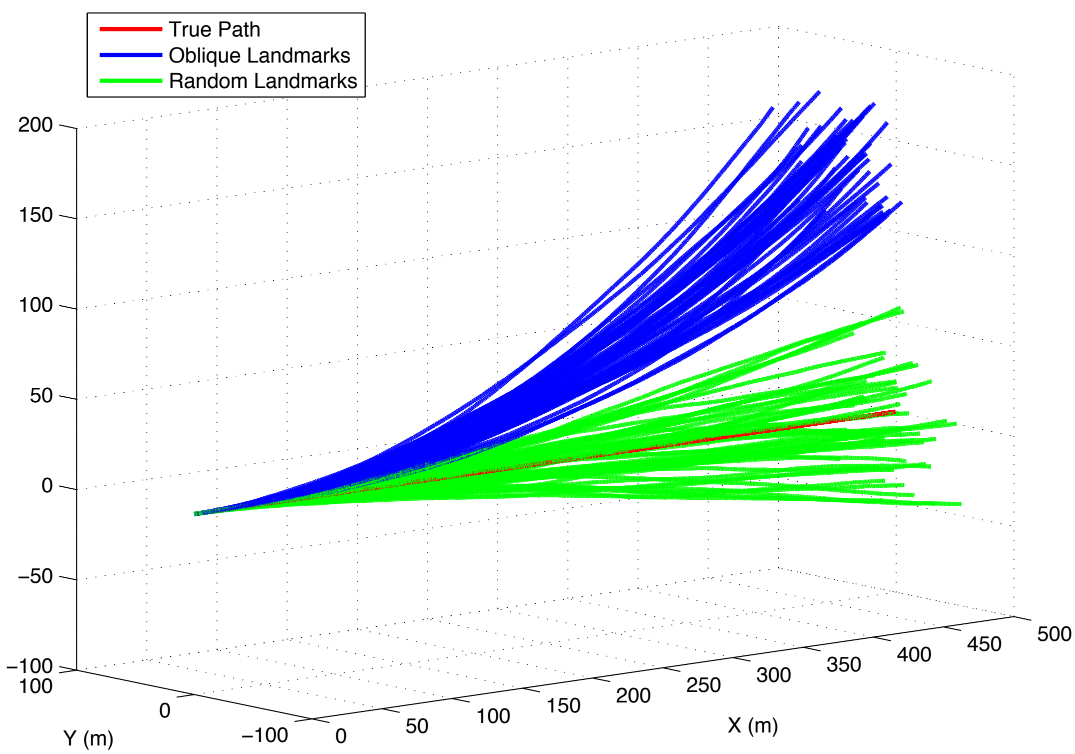


Figure 5.2: Motion estimates from 50 trials, demonstrating the upward VO bias that is introduced when oblique landmarks are used.

As further confirmation of this phenomenon, VO tests were conducted with alternate oblique keypoint configurations; for example, when the landmarks close to the camera were placed on the right hand side of the image and the far away landmarks on the left, the motion estimates showed a left-leaning bias. Similar results were observed for down and right biases, indicating that whichever orientation the landmarks are obliquely presented to the camera, the VO estimate will exhibit a bias toward the far landmarks.

Thus, from these results, we believe that foward-facing stereo-based VO inherently tends to drift in the vertical direction for planetary applications due to the nominally oblique presentation of the terrain to the camera with far landmarks in the upper portion of the image and close landmarks near the bottom. To our knowledge, this phenomenon has not been previously identified in the literature. This could be because the bias is small, and its effect must accumulate over a large sequence of images to produce significant errors in the motion estimate, as was observed in the 10 km traverse shown in Chapter 4. Previous research using relatively short range traverses (Maimone et al., 2007), full Simultaneous Localization and Mapping (SLAM) frameworks (Mei et al., 2010), or IMU data to correct drift (Konolige et al., 2007), may not have observed this bias. Or perhaps it has just been explained away; Konolige et al. (2007), for example, noticed significant VO drifts in a study of 200 m traverse sections, but posit that it is due to incorrect stereo rig calibration and camera tilt deviation.

One related work has been published by Sibley et al. (2007), who identified that stereo cameras are inherently biased to over-estimate the range of landmarks. This bias is caused by the nonlinear nature of the camera model; the estimated range is proportional to $\frac{1}{d}$, where d is the measured keypoint disparity. This $\frac{1}{d}$ nonlinearity in the camera model means that while the disparity measurements may be normally distributed, the resulting spread of range measurements follows a heavy-tailed distribution, as shown representatively in Figure 5.3. The mean of this heavy-tailed distribution is actually farther away than the true range value, producing a bias.

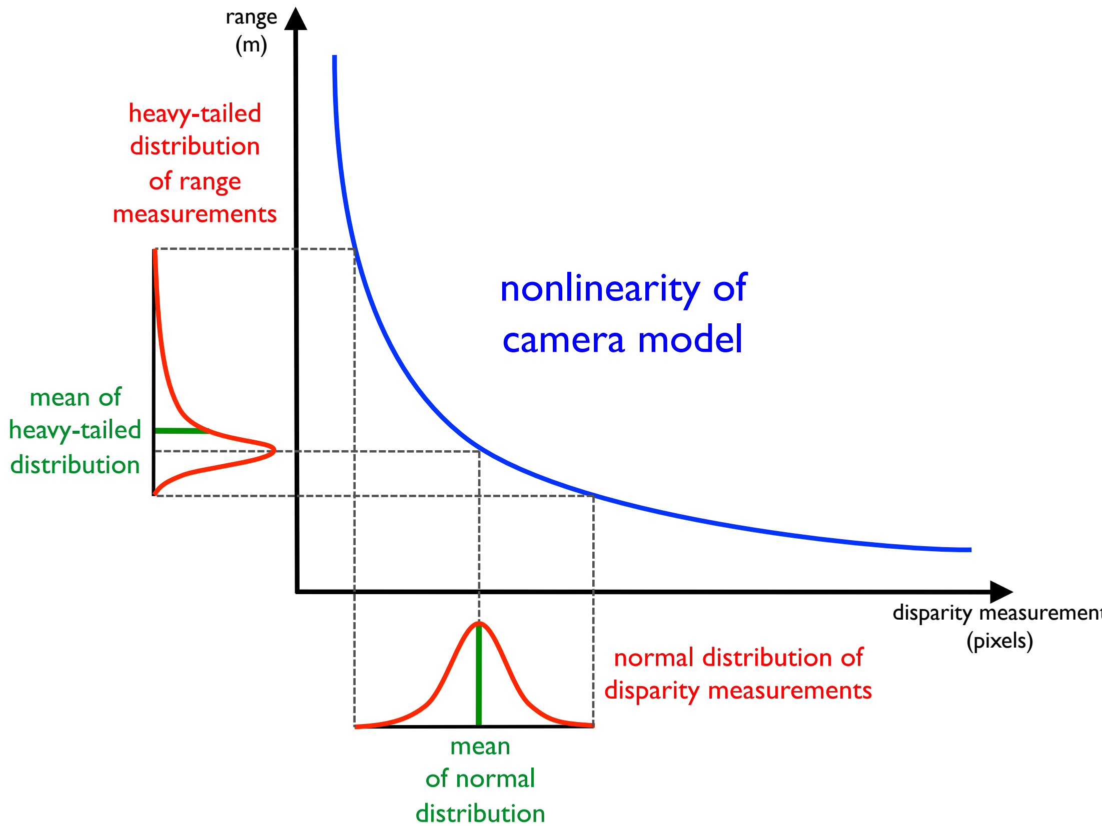


Figure 5.3: An illustration of the range bias for stereo cameras. The mean of the heavy-tailed range distribution is greater than the true range value, producing a bias.

It is important to note from Figure 5.3 that the bias increases with larger range values, as the camera model becomes increasingly nonlinear. This implies that in the case of a planetary rover, there will be more range bias on the landmarks at the top of the image compared with the features at the bottom. We believe that it is this gradient of range biases on the landmarks that produces the upward bias observed in our visual odometry motion estimates.

Given that we know something about the Gaussian noise on the disparity measurements and the nonlinearity of the camera model, it seems plausible that we should be able to estimate this bias. Our approach was to use a technique outlined by Box (1971), which provides a closed-form equation for calculating the bias in nonlinear estimation problems with Gaussian noise on the measurements. However, despite extensive experimentation, this formulation was not able to properly describe the upward bias that we were observing in the VO estimates. After further investigation, we believe that this discrepancy is caused by the inherent disparity thresholds used in stereo visual odometry. These thresholds come into play since there cannot be any negative disparity measurements for matched keypoints, and additionally, a minimum disparity threshold is commonly enforced to prevent very far away landmarks from impeding the nonlinear numerical solution. These thresholds effectively truncate the Gaussian noise distribution on the measurements, which in turn produces a truncated heavy-tailed distribution on the range values. The truncated nature of this distribution disrupts the mean range value; in fact, we have seen that this truncation effect causes underestimation of the range values, rather than the expected overestimation seen by Sibley et al. (2007).

Since the noise on the measurements is not purely Gaussian, the approach described by Box (1971) is no longer appropriate for estimating the bias. We have shown this phenomenon using a simple one-dimensional simulation. The simulation starts with a vehicle on a linear track taking a measurement to some landmark. The vehicle then drives some distance along the track, takes a new measurement of the same landmark, and estimates what the translational distance of travel was. We use a measurement model such that the range is proportional to $\frac{1}{d}$, where d is the value measured by the rover; thus, we introduce the same nonlinearity as in the stereo camera model, and as such, we expect to see similar biases present. The results are shown in Figure 5.4, with each data point being calculated from a million trials at a specific landmark distance. The estimated bias computed with the Box (1971) formulation closely matches the actual bias when Gaussian noise has been applied to the measurements. If, however, the Gaussian noise is truncated, we can see that the actual bias deviates strongly from the estimated bias. This reflects the situation observed for the upward bias in three-dimensional visual odometry.

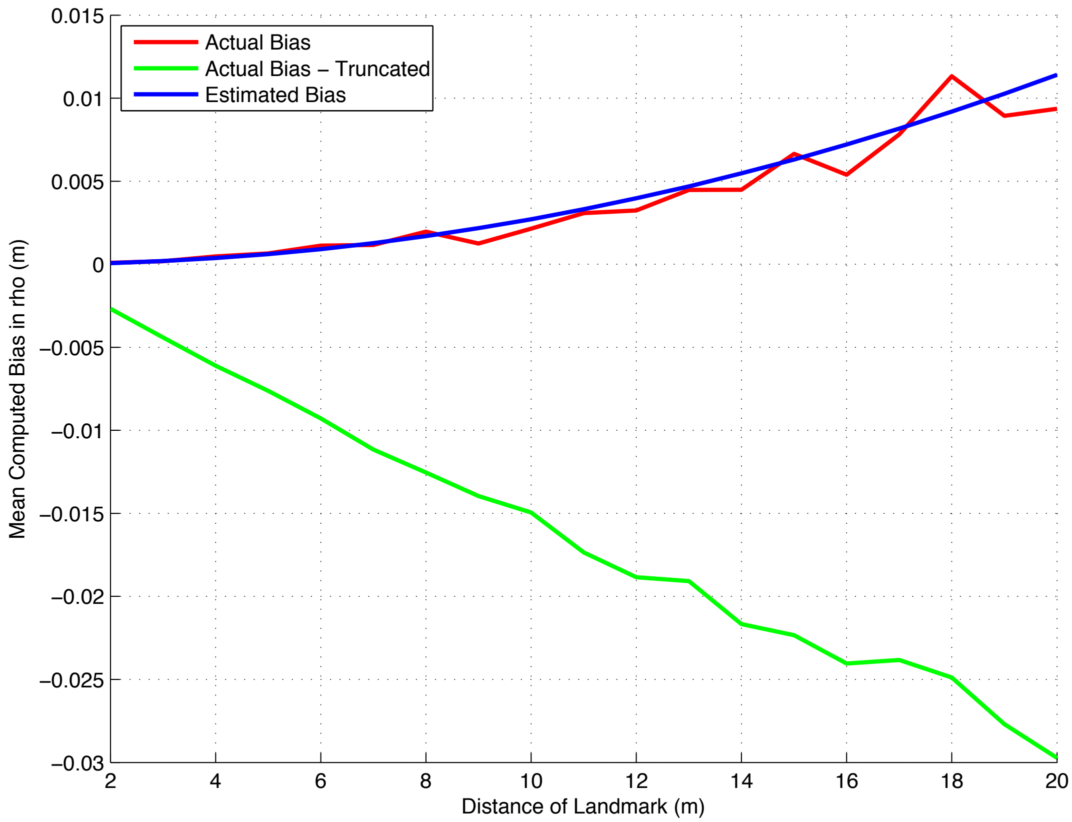


Figure 5.4: Bias results from one-dimensional simulation, averaged the results from a million trials at each landmark distance. The bias estimated using the Box (1971) framework closely approximates the actual bias when using Gaussian noise. However, the bias estimate is poor when the noise is truncated.

Unfortunately, computing an estimate for the upward bias using this truncated Gaussian noise is challenging, and will be the subject of future work. Regardless, the results outlined in this chapter are noteworthy, having identified and largely characterized a bias that affects all planetary visual odometry systems. In lieu of accurate bias estimation, including absolute attitude measurements helps to reduce its effect, as we have observed in Figure 4.13, as does increasing the numbers of frames over which the optimization is performed (Konolige et al., 2007).

Chapter 6

Summary

In this thesis, we have presented a novel algorithm for incorporating sun sensor and inclinometer measurements directly into the visual odometry solution using sparse bundle adjustment. Through rigorous testing on 10 kilometres of data from a planetary analog environment, we demonstrated that these absolute orientation measurements greatly restrict the error growth of the motion estimate, especially over long traversal distances. In particular, the small error achieved by our algorithm over the full traverse loop places it among some of the most accurate visual odometry systems in the world. Importantly, incorporating the sun sensor and inclinometer measurements comes at a very low cost in terms of power, weight, and computation. In other words, one could easily improve an existing visual odometry system with little effort and cost by adding and integrating these two sensors. The mathematical formulation and experimental results presented in this thesis have been published in the literature by Lambert et al. (2011). Looking forward, we plan to investigate methods to further improve the quality of our visual odometry estimates. One such approach would be to use a sliding window approach for visual odometry, including measurements from a handful of previous timesteps into the estimate.

Another contribution of this thesis is the identification the inherent upward bias that affects planetary visual odometry systems. The tendency of VO estimates to travel up into the air has been a mysterious phenomenon, plaguing the results in this thesis and the work of Furgale and Barfoot (2010). Through extensive simulation, however, we have shown definitively that this bias is caused by the oblique presentation of landmarks to the camera. While this inherently affects VO in expansive outdoor environments, such as planetary terrain, it may come into play in other environments as well. For example, Lovegrove et al. (2011) recently published visual odometry results using a rear parking camera which appear to always drift to the right. It may be possible that landmarks close to the camera, like other cars or curbs, tend to appear on one side of the images, producing a bias in the motion estimates. Regardless, being able to accurately estimate this bias in any environment would be an extremely useful contribution. In this thesis, we have characterized the bias in great detail, illustrating the camera model nonlinearity that comes into play and the truncated Gaussian noise that makes estimation difficult. Looking forward, our future research will focus on correctly estimating the upward bias using a truncated noise model, followed by investigating what accuracy gains can be made by subtracting it off. This bias compensation technique could then be combined with the effective error reduction of the sun sensor and inclinometer measurements, producing a robust approach for improving the accuracy of visual odometry for planetary rovers.

In summary, the novel contributions of this thesis are:

1. The first algorithm for directly incorporating sun sensor and inclinometer measurements within a visual odometry pipeline.

2. The first tests of visual odometry (with and without sun sensor and inclinometer measurements) on long-range planetary relevant data.

3. The first identification and characterization of an upward bias in visual odometry for planetary rover operations.

Appendix A

Efficiently Solving Sparse Bundle Adjustment

This appendix is meant to serve as a companion to Section 3.4, outlining the low-level details of solving the sparse bundle adjustment problem in a computationally efficient fashion. We start by recalling equation 3.31, the usual bundle adjustment update step equation:

$$\mathbf{H}^T\mathbf{T}^{-1}\mathbf{H}\begin{bmatrix}\delta\mathbf{x}\\ \delta\mathbf{p}\end{bmatrix} = -\mathbf{H}^T\mathbf{T}^{-1}\mathbf{e}\left(\bar{\mathbf{x}},\bar{\mathbf{p}}\right)$$

In order to derive sparse solution methods for the update step, we start by re-expressing (3.31) as an $\mathbf{A}\mathbf{x}=\mathbf{b}$ problem:

$$\underbrace{\begin{bmatrix}\mathbf{A}_{11} & \mathbf{A}_{12}\\ \mathbf{A}_{12}^T & \mathbf{A}_{22}\end{bmatrix}}_{\mathbf{A}}\underbrace{\begin{bmatrix}\mathbf{x}_1\\ \mathbf{x}_2\end{bmatrix}}_{\mathbf{x}} = \underbrace{\begin{bmatrix}\mathbf{b}_1\\ \mathbf{b}_2\end{bmatrix}}_{\mathbf{b}} \tag{A.1}$$

where

$$\mathbf{A}=\mathbf{H}^T\mathbf{T}^{-1}\mathbf{H},\qquad \mathbf{x}=\begin{bmatrix}\delta\mathbf{x}\\ \delta\mathbf{p}\end{bmatrix},\qquad \mathbf{b}=-\mathbf{H}^T\mathbf{T}^{-1}\mathbf{e}\left(\bar{\mathbf{x}},\bar{\mathbf{p}}\right),$$

and where, recalling from Figure 3.4(c), the $\mathbf{A}_{11}$ and $\mathbf{A}_{22}$ partitions are block-diagonal. The simplest sparse solution method utilizes the Schur complement to manipulate (A.1) into a form that is more efficiently solved. Both sides of (A.1) are premultiplied by

$$\begin{bmatrix} \mathbf{1} & -\mathbf{A}_{12}\mathbf{A}_{22}^{-1} \\ \mathbf{0} & \mathbf{1} \end{bmatrix},$$

producing the following expression:

$$\begin{bmatrix} \mathbf{A}_{11} - \mathbf{A}_{12}\mathbf{A}_{22}^{-1}\mathbf{A}_{12}^T & 0 \\ \mathbf{A}_{12}^T & \mathbf{A}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{b}_1 - \mathbf{A}_{12}\mathbf{A}_{22}^{-1}\mathbf{b}_2 \\ \mathbf{b}_2 \end{bmatrix} \tag{A.2}$$

Note that the solution for $\mathbf{x}_1$ is now decoupled from the solution for $\mathbf{x}_2$, and thus we can write an expression directly for $\mathbf{x}_1$ as follows:

$$\mathbf{x}_1 = \left(\mathbf{A}_{11} - \mathbf{A}_{12}\mathbf{A}_{22}^{-1}\mathbf{A}_{12}^T\right)^{-1}\left(\mathbf{b}_1 - \mathbf{A}_{12}\mathbf{A}_{22}^{-1}\mathbf{b}_2\right) \tag{A.3}$$

Since $\mathbf{A}_{22}$ is block-diagonal, we can solve for $\mathbf{A}_{22}^{-1}$ in an efficient manner. Additionally, the $\left(\mathbf{A}_{11} - \mathbf{A}_{12}\mathbf{A}_{22}^{-1}\mathbf{A}_{12}^T\right)$ is very small, as observed in Figure 3.4(c), so its inverse will not be expensive to compute. Thus, we can solve (A.3) to obtain $\mathbf{x}_1$ in an efficient manner. The expression for $\mathbf{x}_2$,

$$\mathbf{x}_2 = \mathbf{A}_{22}^{-1}\left(\mathbf{b}_2 - \mathbf{A}_{12}^T\mathbf{x}_1\right), \tag{A.4}$$

can also be solved inexpensively, once again owing to the sparsity of $\mathbf{A}_{22}$. Thus, we have solved (A.1) for $\mathbf{x}_1$ and $\mathbf{x}_2$ in an efficient manner, which is what we wanted. However, we may also want to obtain the covariance matrix associated with the state, $\mathbf{x}$. Unfortunately, this covariance matrix is equal to the inverse of $\mathbf{A}$, which is not sparse and will be quite expensive to invert. Thus, this Schur complement method is not the most efficient approach if we require information about the uncertainty of the state. While this was the approach that we used for the experiments in this thesis, we later developed an alternate method which allows us to obtain the covariance matrix in an efficient manner.

This alternate method begins by noting that $\mathbf{A}$ is a symmetric positive definite matrix, and as such, it can be factored as follows through a Cholesky decomposition:

$$\underbrace{\begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{12}^T & \mathbf{A}_{22} \end{bmatrix}}_{\mathbf{A}} = \underbrace{\begin{bmatrix} \mathbf{U}_{11} & \mathbf{U}_{12} \\ \mathbf{0} & \mathbf{U}_{22} \end{bmatrix}}_{\mathbf{U}} \underbrace{\begin{bmatrix} \mathbf{U}_{11}^T & \mathbf{0} \\ \mathbf{U}_{12}^T & \mathbf{U}_{22}^T \end{bmatrix}}_{\mathbf{U}^T} \tag{A.5}$$

It is important to note here that the $\mathbf{U}_{22}$ matrix is also block-diagonal. If we multiply the two $\mathbf{U}$ matrices together, it yields the following expression:

$$\begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{12}^T & \mathbf{A}_{22} \end{bmatrix} = \begin{bmatrix} \mathbf{U}_{11}\mathbf{U}_{11}^T + \mathbf{U}_{12}\mathbf{U}_{12}^T & \mathbf{U}_{12}\mathbf{U}_{22}^T \\ \mathbf{U}_{22}\mathbf{U}_{12}^T & \mathbf{U}_{22}\mathbf{U}_{22}^T \end{bmatrix} \tag{A.6}$$

We can now solve for each of the individual $\mathbf{U}$ elements. We recognize the expression for $\mathbf{A}_{22}$,

$$\mathbf{A}_{22} = \mathbf{U}_{22}\mathbf{U}_{22}^T,$$

as the form of a Cholesky decomposition. Thus, since $\mathbf{A}_{22}$ is block-diagonal, we can solve for $\mathbf{U}_{22}$ efficiently. Turning our attention to the $\mathbf{U}_{12}$ element, we can rewrite the expression for $\mathbf{A}_{12}$ as follows:

$$\mathbf{U}_{12} = \mathbf{A}_{12}\mathbf{U}_{22}^{-T}$$

Since $\mathbf{U}_{22}$ is block diagonal, this expression for $\mathbf{U}_{12}$ is cheap to compute as well. Finally, in order to solve for $\mathbf{U}_{11}$, we rewrite the expression for $\mathbf{A}_{11}$ in the familiar Cholesky form:

$$\left(\mathbf{A}_{11} - \mathbf{U}_{12}\mathbf{U}_{12}^T\right) = \mathbf{U}_{11}\mathbf{U}_{11}^T,$$

While the $\left(\mathbf{A}_{11} - \mathbf{U}_{12}\mathbf{U}_{12}^T\right)$ matrix is dense, as we saw in Figure 3.4(c), $\mathbf{A}_{11}$ is very small. Thus, this Cholesky decomposition will not be expensive to compute. Having determined values for all the elements of $\mathbf{U}$, we can also compute $\mathbf{U}^{-1}$:

$$\begin{bmatrix} \mathbf{U}_{11} & \mathbf{U}_{12} \\ \mathbf{0} & \mathbf{U}_{22} \end{bmatrix}^{-1} = \begin{bmatrix} \mathbf{U}_{11}^{-1} & -\mathbf{U}_{11}^{-1}\mathbf{U}_{12}\mathbf{U}_{22}^{-1} \\ \mathbf{0} & \mathbf{U}_{22}^{-1} \end{bmatrix} \tag{A.7}$$

With all the above terms efficiently computed and at hand, we can now easily compute an expression for $\mathbf{A}^{-1}$:

$$\begin{aligned}
\mathbf{A}^{-1} &= \left(\mathbf{U}\mathbf{U}^T\right)^{-1} \\
&= \mathbf{U}^{-T}\mathbf{U}^{-1} \\
&= \begin{bmatrix} \mathbf{U}_{11}^{-T} & \mathbf{0} \\ -\mathbf{U}_{22}^{-T}\mathbf{U}_{12}^{T}\mathbf{U}_{11}^{-T} & \mathbf{U}_{22}^{-T} \end{bmatrix} \begin{bmatrix} \mathbf{U}_{11}^{-1} & -\mathbf{U}_{11}^{-1}\mathbf{U}_{12}\mathbf{U}_{22}^{-1} \\ \mathbf{0} & \mathbf{U}_{22}^{-1} \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{U}_{11}^{-T}\mathbf{U}_{11}^{-1} & -\mathbf{U}_{11}^{-T}\mathbf{U}_{11}^{-1}\mathbf{U}_{12}\mathbf{U}_{22}^{-1} \\ -\mathbf{U}_{22}^{-T}\mathbf{U}_{12}^{T}\mathbf{U}_{11}^{-T}\mathbf{U}_{11}^{-1} & \mathbf{U}_{22}^{-T}\left(\mathbf{U}_{12}^{T}\mathbf{U}_{11}^{-T}\mathbf{U}_{11}^{-1}\mathbf{U}_{12} + \mathbf{1}\right)\mathbf{U}_{22}^{-1} \end{bmatrix}
\end{aligned} \tag{A.8}$$

Using $\mathbf{A}^{-1}$, we can now solve equation A.1 for our desired update step. However, $\mathbf{A}^{-1}$ also happens to be the covariance matrix associated with the state $\mathbf{x}$. Having already computed this matrix, we can simply pick off $\mathbf{T}_k$, the 6×6 block of $\mathbf{A}^{-1}$ corresponding to the covariance of the rover pose at the current time, k.

Appendix B

Data Tables from Experiments

This appendix catalogues the individual traverse data from the experiments outlined in Chapter 4. Table B.1 presents the visual odometry errors for Harris, FAST, and SURF feature detectors. Table B.2 shows the errors produced using the four permutations of matching and covariance. Table B.3 presents the main result of this thesis, that sun sensor and inclinometer measurements greatly reduce the error of visual odometry. Finally, Table B.4 demonstrates that all but one of the individual traverse estimates travelled up into the air when using regular VO.

Table B.1: VO XYZ error for the Harris, FAST, and SURF feature detectors, expressed as percentage of traversal distance for individual sections.

Section	**Distance (m)**	**Harris (%)**	**FAST (%)**	**SURF (%)**
0	413	10.9	8.1	10.9
1	477	17.6	23.2	13.9
2	606	12.5	15.1	11.4
3	541	11.8	14.3	8.0
4	402	10.5	16.4	6.1
5	552	24.7	31.0	12.5
6	538	19.2	24.3	19.3
7	499	23.0	19.0	7.0
8	444	18.0	42.3	11.2
9	487	14.9	18.8	15.7
10	572	20.4	25.7	16.6
11	386	10.5	16.5	8.2
12	557	33.2	40.0	17.0
13	490	18.9	26.4	17.5
14	442	23.3	30.4	14.6
15	557	16.4	23.4	5.8
16	503	27.3	40.6	20.8
17	423	21.7	30.0	13.0
18	338	14.8	20.0	9.4
19	316	6.8	9.5	5.2
20	296	8.8	7.0	8.6
21	170	3.5	4.3	2.5
22	124	1.3	1.1	1.4

Table B.2: VO XYZ error for uniform matching and uniform covariance (UM, UC), scale matching and uniform covariance (SM, UC), uniform matching and scale covariance (UM, SC), and scale matching and scale covariance (SM, SC), all using SURF features. Error is expressed as percentage of traversal distance for individual sections.

Section	**Distance (m)**	**UM, UC (%)**	**SM, UC (%)**	**UM, SC (%)**	**SM, SC (%)**
0	413	10.9	1.1	5.6	1.6
1	477	13.9	11.1	11.8	5.4
2	606	11.4	2.4	12.4	2.3
3	541	8.0	12.9	10.0	8.1
4	402	6.1	6.2	4.4	5.0
5	552	12.5	12.5	12.2	9.4
6	538	19.3	17.5	15.5	11.3
7	499	7.0	9.8	5.6	10.2
8	444	11.2	11.5	10.3	9.0
9	487	15.7	9.3	12.2	7.6
10	572	16.6	14.5	14.2	10.2
11	386	8.2	3.5	7.8	5.3
12	557	17.0	20.6	14.5	17.5
13	490	17.5	17.2	14.9	14.9
14	442	14.6	18.4	10.1	12.6
15	557	5.8	10.9	5.9	9.0
16	503	20.8	17.8	17.8	15.7
17	423	13.0	15.4	11.2	12.2
18	338	9.4	7.0	8.0	6.5
19	316	5.2	3.9	5.3	4.0
20	296	8.6	7.1	4.8	3.7
21	170	2.5	2.6	7.7	3.0
22	124	1.4	1.3	1.3	1.3

Table B.3: VO XYZ error, with and without sun sensor and inclinometer, expressed as percentage of traversal distance for individual sections.

Section	**Distance (m)**	**VO (%)**	**VO-SS-Inc (%)**
0	413	1.6	1.3
1	477	5.4	2.1
2	606	2.3	0.8
3	541	8.1	1.4
4	402	5.0	1.2
5	552	9.4	0.7
6	538	11.3	2.0
7	499	10.2	1.4
8	444	9.0	1.3
9	487	7.6	1.8
10	572	10.2	1.0
11	386	5.3	0.7
12	557	17.5	1.1
13	490	14.9	1.5
14	442	12.6	1.7
15	557	9.0	1.1
16	503	15.7	3.5
17	423	12.2	1.4
18	338	6.5	1.0
19	316	4.0	1.0
20	296	3.7	1.9
21	170	3.0	2.0
22	124	1.3	1.4

Table B.4: VO altitude error at the end of the traverse, in metres. A positive value indicates that the estimate is above the groundtruth path, while a negative value indicates that it is below.

Section	Distance (m)	Altitude Error (m)
0	413	-3.6
1	477	25.3
2	606	3.8
3	541	42.6
4	402	16.8
5	552	43.8
6	538	30.7
7	499	45.6
8	444	35.5
9	487	22.2
10	572	57.0
11	386	17.3
12	557	91.8
13	490	58.8
14	442	53.4
15	557	34.3
16	503	63.7
17	423	33.2
18	338	17.6
19	316	11.7
20	296	6.7
21	170	2.3
22	124	0.8

Bibliography

Barfoot, T. D., Forbes, J. R., and Furgale, P. T. (2010a). Pose estimation using linearized rotations and quaternion algebra. *Acta Astronautica.* (ref. page 20)

Barfoot, T. D., Furgale, P. T., Stenning, B. E., Carle, P. J. F., Enright, J. P., and Lee, P. (2010.b). Devon Island as a proving ground for planetary rovers. In *Proceedings of the Symposium on Brain, Body, and Machine,* a "Celebration of 25 Years of Research Excellence at the McGill Centre for Intelligent Machines"*, to appear*, Montreal, Quebec. (ref. page 33)

Bay, H., Ess, A., Tuytelaars, T., and Gool, L. V. (2008). Speeded-up robust features (SURF). *Computer Vision and Image Understanding*, 110(3):346 – 359. Similarity Matching in Computer Vision and Multimedia. (ref. page 6)

Box, M. J. (1971). Bias in nonlinear estimation. *Journal of the Royal Statistical Society. Series B (Methodological)*, 33(2):pp. 171–201. (ref. pages 57, 58, and 59)

Brown, D. C. (1958). A solution to the general problem of multiple station analytical stereotriangulation. Technical report, RCP-MTP Data Reduction Technical Report No. 43, Patrick Air Force Base, Florida (also designated as AFMTC 58-8). (ref. pages 10 and 29)

Carle, P., Furgale, P. T., and Barfoot, T. D. (2010). Long-range rover localization by matching lidar scans to orbital elevation maps. *Journal of Field Robotics*, 27(3):344–370. (ref. page 48)

Davison, A., Reid, I., Molton, N., and Stasse, O. (2007). Monoslam: Real-time single camera slam. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(6):1052 –1067. (ref. page 8)

Eisenman, A., Liebe, C., and Perez, R. (2002). Sun sensing on the Mars exploration rovers. In *Aerospace Conference Proceedings, 2002. IEEE*, volume 5, pages 5–2249–5–2262 vol.5. (ref. pages 3 and 11)

Enright, J., Furgale, P., and Barfoot, T. D. (2009). Sun sensing for planetary rover navigation. In *Proceedings of the IEEE Aerospace Conference*, Big Sky, MT. (ref. pages 34 and 48)

Enright, J. and Sinclair, D. (2007). Algorithm enhancements for the SS-411 digital sun sensor. In *Proceedings of the 21st Annual AIAA/USU Conference on Small Satellites*, Logan, Utah. (ref. page 37)

Fischler, M. A. and Bolles, R. C. (1981). Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24:381–395. (ref. page 9)

Furgale, P., Enright, J., and Barfoot, T. (2011a). Sun sensor navigation for planetary rovers: Theory and field testing. *Aerospace and Electronic Systems, IEEE Transactions on*, 47(3):1631 –1647. (ref. page 12)

Furgale, P. T. and Barfoot, T. D. (2010). Visual teach and repeat for long-range rover autonomy. *Journal of Field Robotics,* special issue on "Visual mapping and navigation outdoors", 27(5):534–560. (ref. pages 6, 53, and 61)

Furgale, P. T., Carle, P., Enright, J., and Barfoot, T. D. (2011b). The Devon Island rover navigation dataset. Submitted to the *International Journal of Robotics Research* on October 4, 2010. Manuscript # IJR-10-0896. (ref. page 33)

Harris, C. and Stephens, M. (1988). A combined corner and edge detection. In *Proceedings of The Fourth Alvey Vision Conference*, pages 147–151. (ref. page 5)

Howard, A. (2008). Real-time stereo visual odometry for autonomous ground vehicles. In *Intelligent Robots and Systems, 2008. (IROS 2008). 2008 IEEE/RSJ International Conference on*, pages 3946–3952. (ref. pages 5, 7, and 51)

Hughes, P. C. (1986). *Spacecraft Attitude Dynamics*. John Wiley & Sons, New York. (ref. page 14)

Johnson, A. E., Goldberg, S. B., Cheng, Y., and Matthies, L. H. (2008). Robust and efficient stereo feature tracking for visual odometry. In *IEEE International Conference on Robotics and Automation, 2008.*, pages 39–46, Pasadena, CA. (ref. pages 7, 9, and 50)

Konolige, K., Agrawal, M., and Solà, J. (2007). Large scale visual odometry for rough terrain. In *Proc. International Symposium on Research in Robotics (ISRR)*. (ref. pages 7, 9, 10, 11, 40, 56, and 59)

Konolige, K. and Bowman, J. (2009). Towards lifelong visual maps. In *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, pages 1156 –1163. (ref. page 5)

Lambert, A., Furgale, P., Barfoot, T. D., and Enright, J. (2011). Visual odometry aided by a sun sensor and inclinometer. In *Proceedings of the IEEE Aerospace Conference*, Big Sky, MT. (ref. page 60)

Lee, P., Bunch, T. E., Cabrol, N., Cockell, C. S., Grieve, R. A. F., Rice, J. W., Mckay, C. P., Chutt, J. W., and Zent, A. P. (1998). Haughton-Mars 97 - I: Overview of observations at the Haughton Impact Crater, a unique Mars analog site in the Canadian High Arctic. In *Proceedings of the 29th Lunar and Planetary Science Conference*, pages 1973–1974, Houston, TX. (ref. page 33)

Li, R., Di, K., Howard, A., Matthies, L., Wang, J., and Agarwal, S. (2007). Rock modeling and matching for autonomous long-range mars rover localization. *Journal of Field Robotics*, 24(3):187–203. (ref. page 10)

Lovegrove, S., Davison, A. J., and Ibanez-Guzman, J. (2011). Accurate visual odometry from a rear parking camera. In *Intelligent Vehicles Symposium (IV), 2011 IEEE*, pages 788 –793. (ref. page 61)

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60:91–110. (ref. page 6)

Maimone, M., Cheng, Y., and Matthies, L. (2007). Two years of visual odometry on the Mars exploration rovers. *Journal of Field Robotics*, 24(3):169–186. (ref. pages 2, 7, 9, 10, 11, 50, and 56)

Matthies, L. (1992). Stereo vision for planetary rovers - stochastic modeling to near real-time implementation. *International Journal of Computer Vision*, 8(1):71–91. (ref. page 4)

Matthies, L., Maimone, M., Johnson, A., Cheng, Y., Willson, R., Villalpando, C., Goldberg, S., Huertas, A., Stein, A., and Angelova, A. (2007). Computer vision on Mars. *International Journal of Computer Vision*, 75(1):67–92. (ref. pages 1, 4, and 50)

Matthies, L. and Shafer, S. (1987). Error modeling in stereo navigation. *IEEE Journal of Robotics and Automation*, RA-3(1):239 – 250. (ref. pages 4 and 10)

Mei, C., Sibley, G., Cummins, M., Newman, P., and Reid, I. (2010). RSLAM: A system for large-scale mapping in constant-time using stereo. *International Journal of Computer Vision*, pages 1–17. Special issue of BMVC. (ref. pages 5, 7, 9, 10, 40, and 56)

Moravec, H. (1980). Obstacle avoidance and navigation in the real world by a seeing robot rover. Technical Report CMU-RI-TR-80-03, Robotics Institute, Carnegie Mellon University. (ref. page 4)

Nister, D. (2003). Preemptive ransac for live structure and motion estimation. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 199 –206 vol.1. (ref. page 9)

Nister, D., Naroditsky, O., and Bergen, J. (2006). Visual odometry for ground vehicle applications. *Journal of Field Robotics*, 23(1):3. (ref. pages 7, 8, and 10)

Olson, C. F., Matthies, L. H., Schoppers, M., and Maimone, M. W. (2003). Rover navigation using stereo ego-motion. *Robotics and Autonomous Systems*, 43(4):215–229. (ref. pages 2, 3, and 11)

Rosten, E. and Drummond, T. (2005). Fusing points and lines for high performance tracking. In *IEEE International Conference on Computer Vision*, volume 2, pages 1508–1511. (ref. page 5)

Rosten, E. and Drummond, T. (2006). Machine learning for high-speed corner detection. In *European Conference on Computer Vision*, volume 1, pages 430–443. (ref. page 5)

Sibley, G., Matthies, L., and Sukhatme, G. (2007). Bias reduction and filter convergence for long range stereo. In Thrun, S., Brooks, R., and Durrant-Whyte, H., editors, *Robotics Research*, volume 28 of *Springer Tracts in Advanced Robotics*, pages 285–294. Springer Berlin / Heidelberg. (ref. pages 56 and 58)

Trebi-Ollennu, A., Huntsberger, T., Cheng, Y., Baumgartner, E., Kennedy, B., and Schenker, P. (2001). Design and analysis of a sun sensor for planetary rover absolute heading detection. *Robotics and Automation, IEEE Transactions on*, 17(6):939–947. (ref. page 11)

Volpe, R. (1999). Mars rover navigation results using sun sensor heading determination. In *Intelligent Robots and Systems, 1999. IROS '99. Proceedings. 1999 IEEE/RSJ International Conference on*, volume 1, pages 460–467 vol.1. (ref. page 11)

Wilcox, B. and Nguyen, T. (1998). Sojourner on Mars and lessons learned for future planetary rovers. In *Proceedings of the 28th SAE International Conference on Environmental Systems (ICES)*, Danvers, Massachusetts. (ref. page 11)