Infrastructure Free Rover Localization

A. Nefian, X. Bouyssounousse*, L. Edwards*, M. Dille**, T. Kim**,
E. Hand***, J. Rhizor***, M. Deans*, G. Bebis***, T. Fong*

Carnegie Mellon University, Silicon Valley, CA, USA
e-mail: ara.nefian@nasa.gov

*NASA Ames Research Center, CA, USA
**Stinger Ghaffarian Technologies, CA, USA
***University of Nevada at Reno, USA

Abstract

This paper introduces a system applicable to autonomous planetary exploration rovers that provides localization in the inevitable absence of a Global Positioning System (GPS). Given a terrain elevation map previously constructed from satellite imagery, the proposed method determines rover pose using new imagery from an on-board stereo camera system by combining visual odometry, 3D terrain matching, and horizon visibility refinement. This system is evaluated on experimental data recorded during a 3km traverse using a prototype planetary rover testbed. Results indicate a reduction in localization error by over 60% compared to that derived from wheel odometry alone.

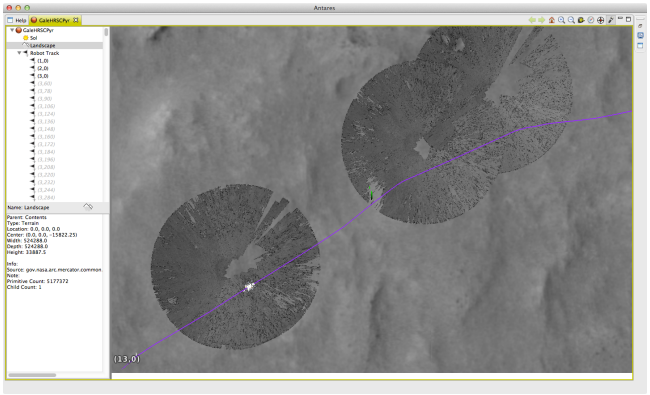

Figure 1. MSL rover panorama localized over HiRISE terrain.

1 Introduction

Planetary rover localization on past and current NASA missions typically relies on tedious manual matching of rover camera views and orbital terrain maps (image and elevation). Alternatively, the location of rovers can occasionally be verified by spacecraft imagery. [1] Figure 1 illustrates several digital elevation models (DEM) reconstructed from the stereo navigation camera (navcam) on-board the Mars Science Laboratory (MSL) rover that have been aligned over the Martian orbital DEM reconstructed from satellite imagery data from the HiRISE mission through stereo image processing. Note in this example that the rover location changes between the predicted path (purple) and the actual location of the rover panoramas. The automated system described in this paper attempts to use the same information (orbital terrain maps and on-board rover imagery) to complement and enhance or possibly eventually completely replace current practices.

A typical approach to image-based pose estimation is to register successive point clouds generated from stereo rover imagery by minimizing their square errors [2, 3] and integrating the relative poses sequentially. To reduce inevitable error accumulation from such incremental pose estimation, [4, 5, 6] most navigation systems take advantage of efficient bundle alignment [7, 8, 9, 10] and make use of Inertial Measurement Unit (IMU) and wheel odometry measurements. [11] Automated horizon matching algorithms have been also proposed to identify the location of a rover using a panoramic surface image and a DEM of the surrounding terrain. [12, 13]

In this paper, we propose a novel unified approach for global pose estimation in the absence of GPS by minimizing a cost function of three sources of errors. The cost function penalizes the errors between 1) the estimated rover pose and the pose predicted through visual odometry, 2) the 3D terrain from the estimated rover pose and the orbital terrain map, and 3) the horizon curve detected in the rover imagery and the horizon rendered from the estimated rover pose over the orbital terrain. A block diagram for the overall system is shown in Figure 2, the components of which are explained in greater detail in the following sections.

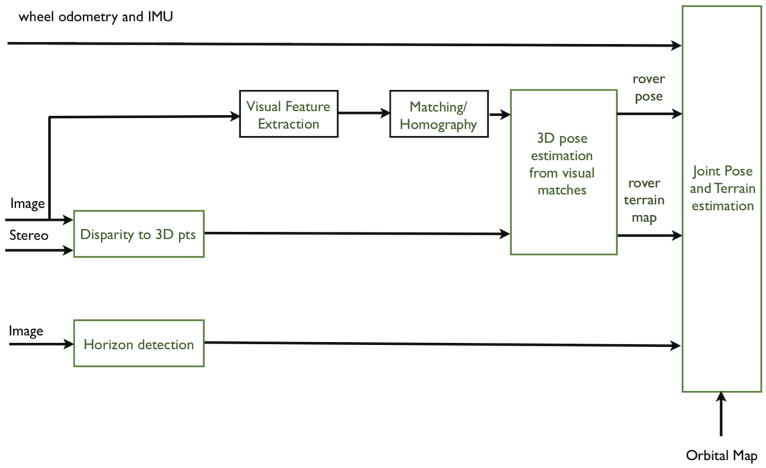

Figure 2. The overall rover localization system.

2 Terrain Reconstruction

The stereo imagery obtained from the camera system [14] on board the rover is used to compute a set of 3D point clouds using the block matching algorithm implemented in the well-known OpenCV package. [15] The stereo camera system with a baseline of 30cm and focal length of 3.5mm is calibrated using the method described by Zhang. [16] Outliers in the reconstructed terrain are removed using a 3×3 morphological filter. Our current system achieves 5Hz on the full image resolution (1388×1038). It is used in visual stereo odometry (Section 3) and supports the localization framework (Section 5) through 3D terrain matching with the terrain model derived from stereo satellite imagery. The reconstructed point clouds are limited to a range of 30m around the rover location since points outside this range have a spatial resolution below the resolution of the orbital terrain model and often correspond to noisy data. Figure 3 shows an example left camera image of the stereo system and its corresponding disparity map.

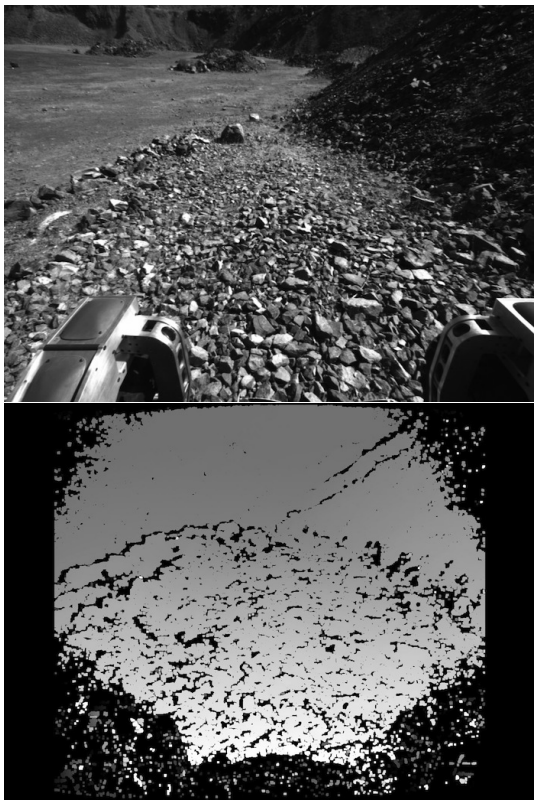

Figure 3. Left camera image of the stereo system (top) and corresponding disparity image (bottom).

3 Stereo Visual Odometry

Stereo visual odometry re-estimates the rover pose at every frame using the rover imagery and the point clouds generated as described in Section 2. The stereo visual odometry system starts with the detection of SURF keypoints. [17] SURF keypoint extraction is faster than using SIFT [18] features and allows for a larger and more reliable set of matches than may be obtained using either BRISK [19] or ORB [20] keypoints. The keypoints extracted from the current image are matched against images from the previous frame using the FLANN fast approximate nearest-neighbor library. [21] To reduce the number of false matches, the process is repeated by matching keypoints in the previous frame to keypoints in the current frame and selecting only those keypoints and matches that pass both conditions. Additional remaining outlier matches are removed using the RANSAC algorithm [22] and by constraining a homographic transformation be-

tween matched keypoints in consecutive frames. The current rover pose relative to its previous pose is estimated using the matched keypoints and their associated 3D position (Section 2) with the method described by Umeyama. [3] The output of stereo visual odometry pose is used as one input to the localization framework described in Section 5. Figure 5 shows two sections of a 3D and texture mapped terrain model built using our stereo visual odometry approach.

Figure 4. Visual feature matches between the current (right) and previous (left) frame using BRISK (top) and SURF (bottom) features.

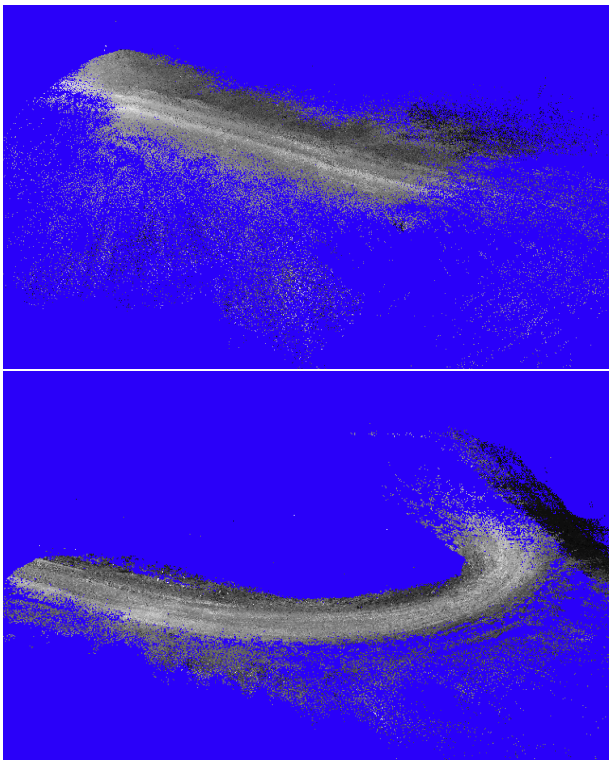

Figure 5. Rover map obtained through visual odometry.

4 Horizon Matching

Beyond the range in which terrain can be reconstructed from stereo imagery, visual features from satellite imagery and rover imagery are often difficult to match due to variations in image resolution, camera viewing angle, terrain illumination, and unknown local albedo. However, the horizon curve in rover images frequently remains a discriminant visual feature that enables global pose estimation by matching with the rendered horizon obtained from the orbital terrain map. The next subsections describe the horizon detection and rendering techniques used in our approach.

4.1 Horizon Detection

The statistical approach for horizon detection described in this paper does not use any specific knowledge of local terrain or sky intensity distribution and is meant to perform well on a large variety of images collected from Earth, Mars, the moon, or other planetary bodies. Our method uses a Bayesian image formation model and a set of visual observations to determine the best image segmentation into ground and sky areas and thereby determine the most likely horizon curve.

A pixel at position (i, j) in the image (row i and column j) is associated with an observation (visual feature) O_{ij} and a hidden node q_{ij}. The hidden nodes have an unknown binary value corresponding to lying on either sky or ground segments. The dense (per pixel) visual features used for horizon detection consists of the pixel gray scale intensity and local feature density. The gray scale intensity image was chosen to support the development of a color-independent method usable in various illumination conditions and planetary environments. The local feature density is computed as the ratio of the number of local edges within a local window around each pixel and the area (in pixels) of the local window. In our particular implementation the edges are computed using a 3×3 Sobel edge detector and the size of local window is chosen to be 7×7 pixels.

Let $P(O_{ij}|q_{ij})$ be the distribution of observation vector O_{ij} at pixel ij in the image given the binary value (g:ground or s:sky) of the corresponding hidden node q_{ij}. Figure 6 illustrates the initial distribution of visual features over ground (ground 1 and 2) and sky (clear sky and cloudy). High-intensity areas (low and high feature density) are associated with cloud regions. High-intensity features and low feature density area are associated with

sky. Low-intensity features having low density are associated with terrain in heavy shadows (ground 2), while low-intensity features in high-density areas are associated with regularly lit terrain.

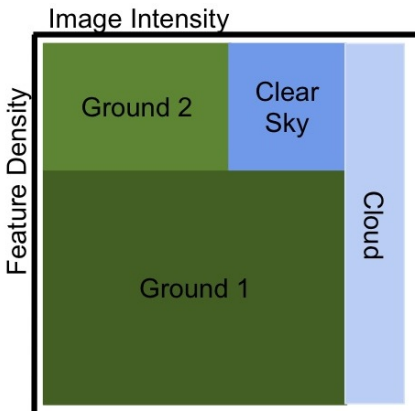

Figure 6. $P(O_{ij}|q_{ij})$ sky (clear sky and cloudy sky) and terrain (ground 1 and 2) distribution over the space of visual features (image intensity and feature density).

Let H be the height of the image and $P(\mathbf{O}_j|\mathbf{Q}_j)$ be the probability of the observation vectors $\mathbf{O}_j = O_{1j} \dots O_{Hj}$ in image column j given the corresponding sequence of hidden nodes $\mathbf{Q}_j = q_{1j} \dots q_{Hj}$. For observation vectors extracted by scanning each column from top to bottom, we can reasonably assume that there is only one transition from sky to ground and none from ground to sky. Therefore, the observation probability in column j given that the transition from sky to ground occurs in pixel at position (k, j) is given by the following equation:

$$P(\mathbf{O}_j|\mathbf{Q}_j) = \prod_{i=1}^{k} P(q_{ij} = s) \prod_{i=k+1}^{H} P(q_{ij} = g). \qquad (1)$$

There are a total H values associated with each $\mathbf{Q}_j$ sequence, one for each pixel in the column j where the unique transition from sky to terrain occurs. The pixel location that achieves the highest value in Equation 1 is the horizon pixel in column j. The set of all the horizon pixels in each column defines the horizon line. The approach described here assumes that each column is independent of its neighbors, which can generate large variations in the horizon line from column to column that often does not correspond to natural environments. A post-processing step to smooth the resulting curve is thus applied to generate the final horizon line.

Figure 7 illustrates an example of the horizon curve detection (blue) in a rectified image captured by an onboard camera.

Figure 7. Example of horizon curve (blue) detection over a rectified navcam image.

4.2 Horizon Rendering

Horizon rendering generates a synthetic view of the horizon as would be observed by the rover-mounted camera. This view is based on the orbital DEM and the camera intrinsic and extrinsic calibration. An accurate rendered horizon curve is often determined by remote topographical features captured in large coverage orbital terrain models. However, handling very large terrain surfaces at the resolution required for terrain matching can easily exceed computer memory limits. This is of particular concern when the processing is offloaded from the host computer to a Graphical Computational Unit (GPU) for fast rendering. To accommodate these constraints, a low-coverage (.8×.8 km), high-resolution (1 m per post) terrain model is augmented with a large-coverage (10×10 km) low-resolution terrain (9 m per post). The resulting terrain model is split into a set of tiles with multiple subsampled resolution levels. This approach satisfies both the wide coverage requirements for horizon rendering and high resolution requirements for terrain matching, while accommodating the memory constraints of a typical GPU (around 1 GB for our processors). The rendered image is computed using standard OpenGL libraries, [23] and the horizon curve is computed as the boundary of the rendered surface. Figure 8 illustrates a rendered image as seen by the left camera of the stereo system on board the rover and from a known rover pose. The red area denotes the surface rendered by using the large coverage, low resolution orbital terrain model. This area would not be visible if the only the low-coverage, high-resolution terrain were used in rendering, leading to incorrect horizon curve rendering.

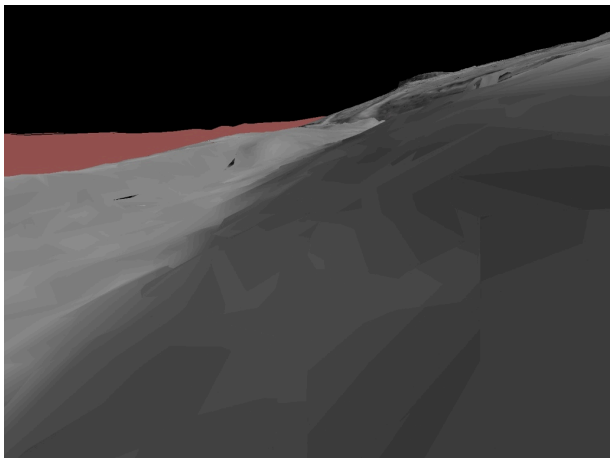

Figure 8. Example of a rendered horizon image depicting both local (gray) and wide-area (red) models.

5 Localization Framework

The output of each of the preceding components is used as a contribution to an *overall* rover pose as now described. First, wheel odometry is obtained through a typical Extended Kalman Filter (EKF) fusing IMU, steering, and wheel rotation measurements. The incremental joint wheel and visual odometry (Section 3) estimate is used as an initial seed for finding the rover pose within the predetermined orbital map.

Rover localization—finding the optimal rotation ($\mathbf{R}$) and translation ($\mathbf{T}$) from a global reference point—is formulated as a cost function $\mathbf{L}$ minimization problem,

$$\{\tilde{\mathbf{R}}, \tilde{\mathbf{T}}\} = \arg\min_{\{\mathbf{R},\mathbf{T}\}} \mathbf{L}(\mathbf{R}, \mathbf{T}).$$

The localization cost function $\mathbf{L}(\mathbf{R}, \mathbf{T})$ is given by

$$\mathbf{L}(\mathbf{R}, \mathbf{T}) \quad = \quad \omega_h(\mathbf{h}_d - h_r(\mathbf{R}, \mathbf{T}))^2 + \omega_e(\mathbf{e}_o - \mathbf{e}_r(\mathbf{R}, \mathbf{T}))^2,$$

where $\mathbf{h}_d$ and $\mathbf{h}_r(\mathbf{R}, \mathbf{T})$ are the detected and rendered horizon curves (Section 4) respectively. $\mathbf{e}_o$ is the terrain elevation of the orbital map, and $\mathbf{e}_r(\mathbf{R}, \mathbf{T})$ is the terrain elevation determined from the point clouds obtained from the rover stereo camera (Section 2) and the rover estimated global position. $\mathbf{x}(\mathbf{R}, \mathbf{T})$ is the global position determined by the estimated global rotation and translation. The weights ω_h and ω_e are chosen such that $\omega_h + \omega_e = 1$ and represent the reliability of each of the horizon and terrain elevation modules used in localization respectively. These weights are also chosen to normalize with various number of samples and vector sizes in each modality.

With the current system, the rover localization within the orbital map cannot be computed at every frame due to its high computational complexity. To provide a trade-off between localization accuracy and run-time requirements in current experiments, the overall rover pose is computed every 300 frames with odometry updates available at each image frame.

6 Experimental Results

Data used to validate the described algorithms were gathered from a 3km traverse of a prototype planetary rover operating in the Basalt Hills State Park, California at an average speed of 0.8m/s. Images from the stereo camera system on-board the rover described in Section 2 were captured at an average 2Hz. Figure 9 shows a typical reduction in localization error obtained using the proposed system (labeled "advanced nav") as compared to errors introduced by using a purely wheel odometry-based localization solution. These errors are calculated with reference to ground truth data obtained by a high-end GPS-INS system operating concurrently. Large steps in localization error visible in this plot are due to the aforementioned periodic invocation of the proposed system due to runtime resource constraints. On average, localization error is reduced by over 60% compared to the wheel odometry solution.

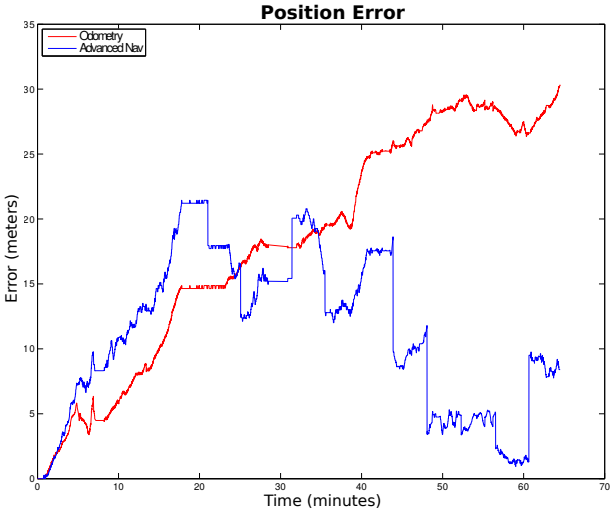

Figure 9. Localization errors for the proposed method (blue) and wheel odometry (red) vs true GPS position.

Figure 10 shows an overhead view of the rover traverse ground truth determined by GPS (green) compared against the estimated trajectories using wheel odometry (red) and the proposed solution (blue). Note that the differences between the GPS and wheel odometry trajectories increase from the starting point (top right in Figure 10) due to inevitable incremental error accumulation, while the proposed solution that periodically incorporates

absolute measurements remains significantly closer to the GPS track during the entire traverse.

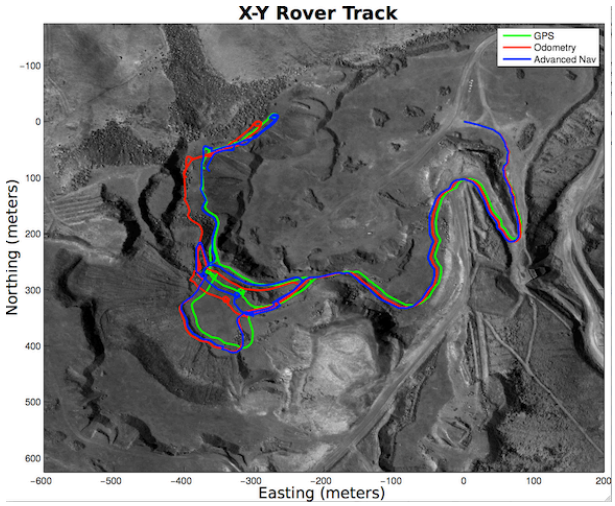

Figure 10. Estimated rover track using the proposed method (blue), wheel odometry (red), and GPS (green).

7 Conclusions

The localization method presented in this paper combines several vision-based techniques—visual odometry, terrain matching, and horizon visibility refinement—using data from both live local rover and existing global orbital views to accurately determine the rover location within an orbital map. The accuracy of the method makes it suitable for autonomous or semi-autonomous robotic planetary exploration in the absence of a pre-existing GPS or other beacon-based infrastructure. Future work will be directed towards improved real-time implementation of the localization system and testing with various rover configurations in a variety of terrestrial environments, as well as with imagery returned from current Mars missions.

References

[1] Rongxing Li, Shaojun He, Yunhang Chen, Min Tang, Pingbo Tang, Kaichang Di, Larry Matthies, Raymond E Arvidson, Steven W Squyres, Larry S Crumpler, et al., "MER Spirit rover localization: Comparison of ground image–and orbital image–based methods and science applications," *Journal of Geophysical Research: Planets (1991–2012)*, vol. 116, no. E7, 2011.

[2] AJ Stoddart and A Hilton, "Registration of multiple point sets," in *Pattern Recognition, 1996., Proceedings of the 13th International Conference on*. IEEE, 1996, vol. 2, pp. 40–44.

[3] Shinji Umeyama, "Least-squares estimation of transformation parameters between two point patterns," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 13, no. 4, pp. 376–380, 1991.

[4] Sebastien Gemme, David Gingras, Alessio Salerno, Erick Dupuis, François Pomerleau, and François Michaud, "Pose refinement using ICP applied to 3-D LIDAR data for exploration rovers," in *Proc. 2012 International Symposium on Artificial Intelligence, Robotics and Automation in Space*, 2012.

[5] Fabio Cozman, Eric Krotkov, and Carlos Guestrin, "Outdoor visual position estimation for planetary rovers," *Autonomous Robots*, vol. 9, no. 2, pp. 135–150, 2000.

[6] Luca Lucchese, Gianfranco Doretto, and Guido M. Cortelazzo, "A frequency domain technique for range data registration," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 11, pp. 1468–1484, 2002.

[7] Kurt Konolige and Willow Garage, "Sparse sparse bundle adjustment.," in *BMVC*, 2010, pp. 1–11.

[8] Niko Sünderhauf and Peter Protzel, "Towards using sparse bundle adjustment for robust stereo odometry in outdoor terrain," *Towards Autonomous Robotic Systems*, pp. 206–213, 2006.

[9] Michael Kaess, Ananth Ranganathan, and Frank Dellaert, "iSAM: Incremental smoothing and mapping," *Robotics, IEEE Transactions on*, vol. 24, no. 6, pp. 1365–1378, 2008.

[10] Michael Kaess, Hordur Johannsson, Richard Roberts, Viorela Ila, John J Leonard, and Frank Dellaert, "iSAM2: Incremental smoothing and mapping using the bayes tree," *The International Journal of Robotics Research*, vol. 31, no. 2, pp. 216–235, 2012.

[11] Rongxing Li, Kaichang Di, Larry H Matthies, Raymond E Arvidson, William M Folkner, and Brent A Archinal, "Rover localization and landing-site mapping technology for the 2003 mars exploration rover mission," *Photogrammetric engineering and remote sensing*, vol. 70, no. 1, pp. 77–90, 2004.

[12] Clark F Olson and Larry H Matthies, "Maximum likelihood rover localization by matching range maps," in *Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on*. IEEE, 1998, vol. 1, pp. 272–277.

[13] EE Palmer, RW Gaskell, LD Vance, MV Sykes, BK McComas, and WC Jouse, "Location identification using horizon matching," in *Lunar and Planetary Institute Science Conference Abstracts*, 2012, vol. 43, p. 2325.

[14] Allied Vision Technologies, "Gige vision camera with Sony ICX267 and PoE option," .

[15] Gary Bradski, "The OpenCV library," *Doctor Dobbs Journal*, vol. 25, no. 11, pp. 120–126, 2000.

[16] Zhengyou Zhang, "Flexible camera calibration by viewing a plane from unknown orientations," in *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*. IEEE, 1999, vol. 1, pp. 666–673.

[17] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool, "SURF: Speeded up robust features," in *Computer Vision–ECCV 2006*, pp. 404–417. Springer, 2006.

[18] David G Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[19] Stefan Leutenegger, Margarita Chli, and Roland Siegwart, "BRISK:Binary Robust Invariant Scalable Keypoints," *International Conference of Computer Vision*, 2011.

[20] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski, "ORB: an efficient alternative to SIFT or SURF," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 2564–2571.

[21] Marius Muja and David G Lowe, "Fast approximate nearest neighbors with automatic algorithm configuration.," in *VISAPP (1)*, 2009, pp. 331–340.

[22] Martin A Fischler and Robert C Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[23] Dave Shreiner and The Khronos OpenGL ARB Working Group, *OpenGL Programming Guide: The Official Guide to Learning OpenGL, Versions 3.0 and 3.1*, Addison-Wesley Professional, 7th edition, 2009.