AstroNav: Robust, High Rate SLAM for Planetary Exploration

Kerry Snyder
kerry.snyder@
astrobotic.com

Eric Amoroso
eric.amoroso@
astrobotic.com

Andrew Horchler
andrew.horchler@
astrobotic.com

Fraser Kitchell
fraser.kitchell@
astrobotic.com

Astrobotic Technology Inc.
2515 Liberty Ave.
Pittsburgh, PA 15222

ABSTRACT

Astrobotic is developing a robust, high-rate simultaneous localization and mapping (SLAM) system under a NASA STTR. This work provides visual and LIDAR-based navigation in dark and GPS-denied environments for the purposes of cave entry, mapping, and exit. Unique challenges include highly dynamical motion, multiple transitions between light and dark environments, limited computing, and safe, energy efficient operation in caves. We achieve under 1.5% linear drift and operation at up to 7.4x real-time.

1. SYSTEM DESIGN

1.1 Hardware

Astrobotic's stereo visual-inertial navigation software assembles COTS technologies, first by integrating two USB2 mvBlueFOX-MLC200w grayscale global shutter 752x480 cameras. The cameras are mounted on a 30cm carbon fiber baseline with wide-FOV S-mount 2.0mm lenses, providing a 180 degree FOV. A VectorNav VN-100 MEMS IMU is centrally mounted and also interfaced through USB (see Figure 1). Sensor logging and navigation is performed on a Gigabyte BRIX Mini-PC with a dual-core i7 processor and 16GB of RAM.

1.2 Software

Visual landmarks are matched between the left and right camera and tracked over time using pyramidal optical flow. Inertial observations temporally connect consecutive camera observations and predict the pose of new camera observations over short timescales using preintegration[4]. All observations are combined with iSAM2[5], an incremental smoothing backend. This Bayes tree formulation continuously and efficiently optimizes the current state estimate as a nonlinear bundle adjustment system.

AstroNav is implemented in C++ using OpenCV[2] and GTSAM[3] and interoperates with ROS[7] for data logging and visualization. A multi-threaded pipeline performs feature initialization, optical flow, and local windowed bundle adjustment in real-time. The frontend is periodically synchronized with the iSAM2 mapping backend to maintain accuracy and speed.

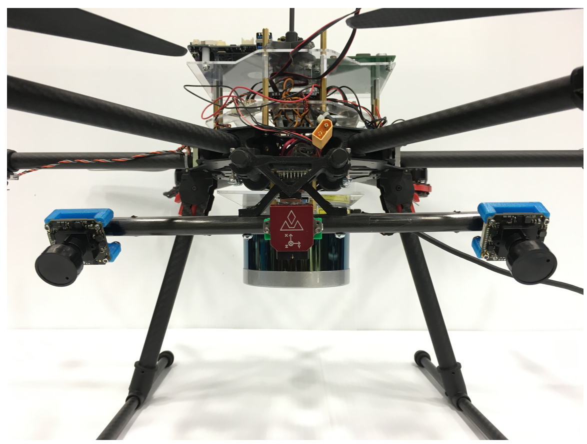

Figure 1: Hexcopter stereo and LIDAR system.

2. PERFORMANCE AND TESTING

This system operates at up to 7.4x real time on commodity computing and has shown drift rates below 1.5% over ten- and hundred-meter-long paths under a variety of lighting and motion conditions. Multiple test datasets are presented below. In testing, performance is quantified using AprilTags[6], but no external infrastructure or initialization procedure is required for operation.

3. ONGOING WORK

For dark and underground operation, Astrobotic is integrating a Velodyne VLP-16 Puck LIDAR sensor (see Figure 1). LIDAR processing is based on LOAM[9] and blam[1], with a combination of ICP based point cloud matching and feature based alignment using the Point

Table 1: Current AstroNav performance

Dataset	Length (m)	Linear Drift %	Angular Drift (deg/m)	Speedup (vs. real-time)
floorplan	163.1	1.55	0.004	6.0
updownstairs	132.9	1.04	0.037	6.0
highbay	87.5	0.50	0.105	5.3
50ft	35.2	0.69	0.033	6.9
stairs	146.8	0.01	0.076	7.4
oneloop	48.8	0.44	0.058	3.9
threeloop	118.9	0.13	0.139	4.1

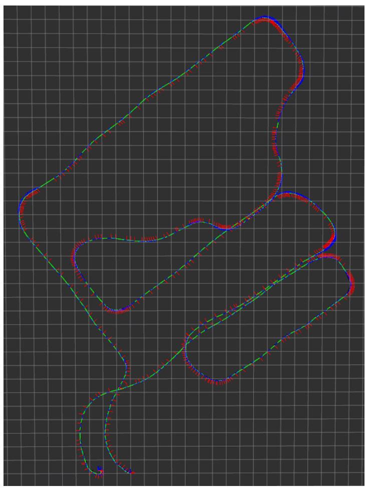

Figure 2: Top view of floorplan dataset with R/G/B X/Y/Z axes. Each square is 1m.

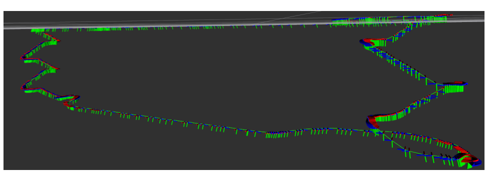

Figure 3: Side view of updownstairs dataset, which travels down and then back up one story, with R/G/B X/Y/Z axes.

Cloud Library[8]. LIDAR observations are integrated as a relative pose measurement into the incremental smoother backend.

Further extensions under development include loop closure detection, dense modeling, monocular operation, and hardware acceleration.

4. ACKNOWLEDGMENTS

This work has been funded under Phases I and II NASA STTR, contract number NNX15CK15P, in collaboration with Wennie Tabib, William "Red" Whittaker, and Nathan Michael.

5. REFERENCES

[1] B(erkeley) l(ocalization) a(nd) m(apping). https://github.com/erik-nelson/blam. Accessed: 2017-02-05.

[2] G. Bradski. Opencv. *Dr. Dobb's Journal of Software Tools*, 2000.

[3] F. Dellaert. Factor graphs and gtsam: A hands-on introduction. Technical report, Georgia Institute of Technology, 2012.

[4] C. Forster, L. Carlone, F. Dellaert, and D. Scaramuzza. Imu preintegration on manifold for efficient visual-inertial maximum-a-posteriori estimation. Georgia Institute of Technology, 2015.

[5] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. J. Leonard, and F. Dellaert. isam2: Incremental smoothing and mapping using the bayes tree. *The International Journal of Robotics Research*, 31(2):216–235, 2012.

[6] E. Olson. Apriltag: A robust and flexible visual fiducial system. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 3400–3407. IEEE, 2011.

[7] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng. Ros: an open-source robot operating system. In *ICRA workshop on open source software*, volume 3, page 5. Kobe, 2009.

[8] R. B. Rusu and S. Cousins. 3D is here: Point Cloud Library (PCL). In *IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, May 9-13 2011.

[9] J. Zhang and S. Singh. Loam: Lidar odometry and mapping in real-time. In *Robotics: Science and Systems*, volume 2, 2014.