Resource-Limited Exploration Autonomy for Planetary Rovers

Varsha Kumar
Mentored by Dr. Red Whittaker

ABSTRACT:

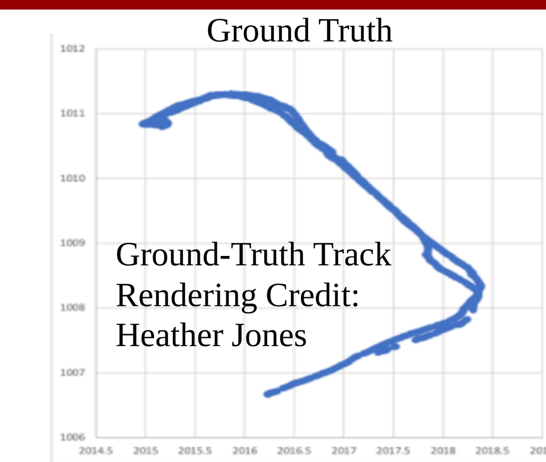
Robot functionality in truly uncontrolled environments is thwarted by the debility of current autonomy software. Carnegie Mellon's ice-finding lunar micro-rover will be humanity's first real moonshot that compels ambitious autonomy from minimal computing and sensing in an unpredictable and highly uncontrolled environment. MoonRanger design constraints, such as limited compute power and limited sensing, compel unprecedented autonomous functionality that will rely on algorithms for determining location, orientation, and maps of surrounding terrain. These elements of autonomy are keystones in MoonRanger's overarching architecture, which, by necessity, encompasses both software and hardware considerations. This research details the foundations of a robust pose estimation under these limitations. The investigation evinces how preliminary algorithm application on experimental data is informing decisions as to the validity of design. The poster projects future research and speculates MoonRanger's impact of shattering the boundaries of autonomous robotics.

Image Credit: Uland Wong [12]

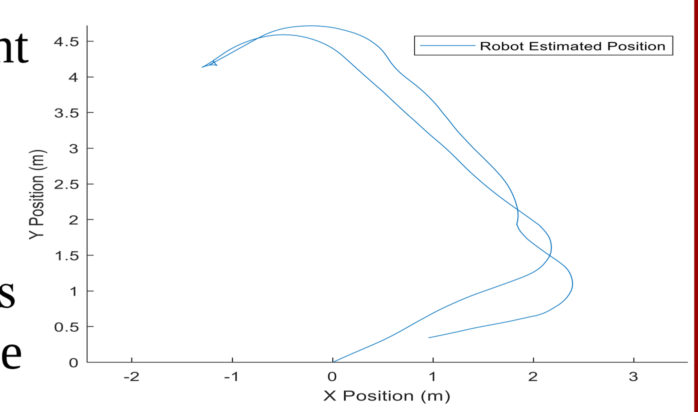
BACKGROUND AND CHALLENGES:

As microrovers travel to distant goals, they go out of communication with their landers and earth. The critical requirement is to autonomously to return to the lander. Without excellent navigation, any rover would remain out of communication range and be lost forever. Given that the rover referenced in this research will search for ice at a lunar pole, the grazing sunlight presents truly black shadows (as pictured above) and many dark lighting conditions. As such, typical Visual-Inertial methodologies for Pose Estimation will fail. The need is for highly innovative solutions that combine data from non-visual sensors, sun sensors, and laser-lit image-processed terrain stripes. Additionally, there is a need for a vision-denied backup pose estimate that can guide such rovers home in the eventuality of camera electronic failure. Given the computational constraints of small microrovers and need for utilizing error-correcting code on the processor, other Autonomy Software components, such as mapping algorithms, must be streamlined and will rely heavily on a robust, efficient pose estimate.

EXPERIMENTAL EVALUATION OF VISION-DENIED POSE ESTIMATION:



Below: Extended Kalman Filter Pose Estimate, Integrating Encoders and IMU



Pictured on the right is a comparison of Ground Truth rover pose (top) and a pose estimate using an Extended Kalman Filter (EKF) with both Inertial Measurement Unit (IMU) and encoders (bottom). The EKF integrates estimates from on-board encoders and IMU. The EKF used was ROS's robot_pose_ekf package[5].

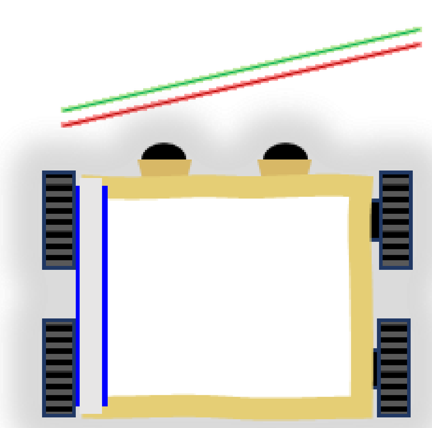
Sole reliance on the EKF with IMU and encoder values drifts significantly. The error in the example trek was roughly 5 percent of distance traveled. Proportionately, for a trek that is one kilometer out and a kilometer back, (MoonRanger's ambition), the error would be roughly 100 meters. This is larger than the lander's guaranteed communication range and would lead to loss of communication, hence loss of the rover. For this reason, some form of visual odometry that succeeds in darkness is absolutely essential.

LINE-STRIPING LOCALIZATION:

Avionics and Image Credit: Haidar Jamal, Vaibhav Gupta

Conventional Visual-Inertial (VI) Algorithms fail at the lunar pole due to darkness. Alternately, laser line striping is evident in darkness as a form of structured light perception. An additional advantage of processing on laser line stripes is a drastic efficiency increase relative to conventional VI solutions by restricting the search space of points to match. The implementation and algorithm follow.

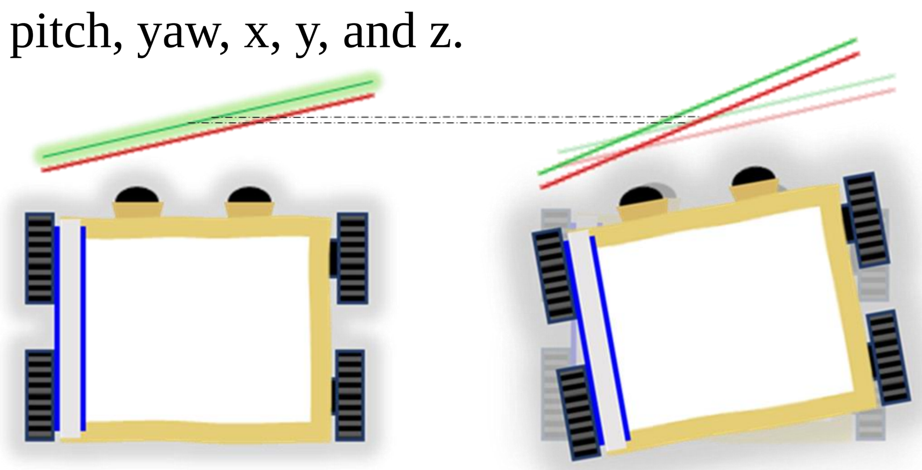
<u>Implementation</u>



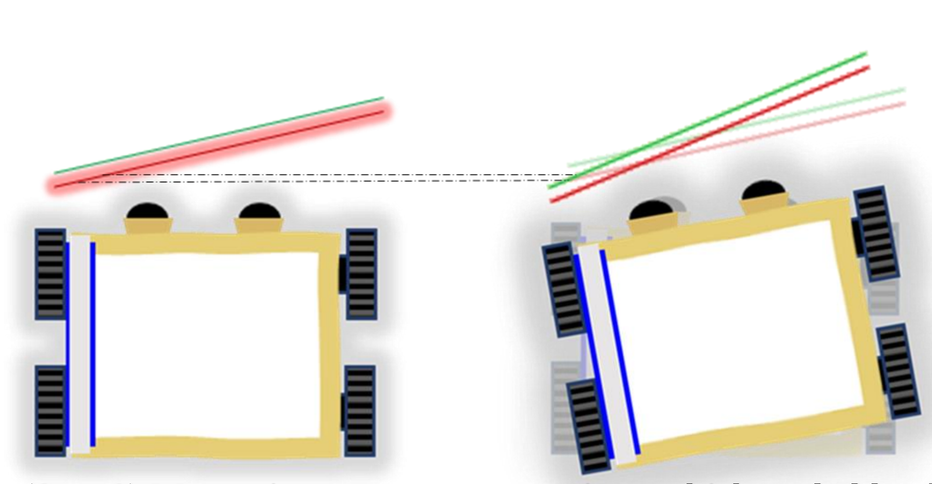
Shine a laser at the ground so that it lights a line of terrain. Take a one picture with lasers on and one with lasers off. Difference the lit and unlit images (on a particular color channel) to accentuate the line. Image process to get crisp lines. Use geometry-based stereo to reconstruct points along the line [13]. Pictured at the top are transverse red stripes. For the algorithm that follows, the configuration (right) will project one green and one red parallel, diagonal stripe ahead of the rover. Distance between the stripes is equal to the v_r/t_l, where v_r is rover velocity when driving straight forward and t_l is the time between laser line stripe point reconstructions, on average. Two colors are used to eliminate the computational processing needed to determine which stripe is farther from the rover.

<u>Algorithm:</u>

(1) As the rover moves, update and store the point of rotation that corresponds geometrically and kinematically to the motion of the line stripes as a result of the rotational motion, as measured by the rover's Inertial Measurement Unit.
(2) At the receipt of each new pair of laser line stripe point reconstructions, randomly select k points along the new red line stripe, on the correct side of the point of rotation, and match to the old green laser line stripe, thickened by fitting a gaussian to each point. The matching point is the point with the closest height on the thickened green laser line stripe within a radius equal to $\alpha*(v_r/t_l)$. (Illustrated below, left)
(3) Repeat Step 2 for the green laser line stripe. (Illustrated below, right)
(4) Compute the rigid transform for each tuple of matched points.
(5) Return the rigid transform created by selecting the individual medians of changes roll, pitch, yaw, x, y, and z.



(Step 2) Map points on new red stripe to thickened old green stripe on the correct side of the point of rotation.



(Step 3) Map points on new green stripe to thickened old red stripe on the opposite, correct side of the point of rotation.

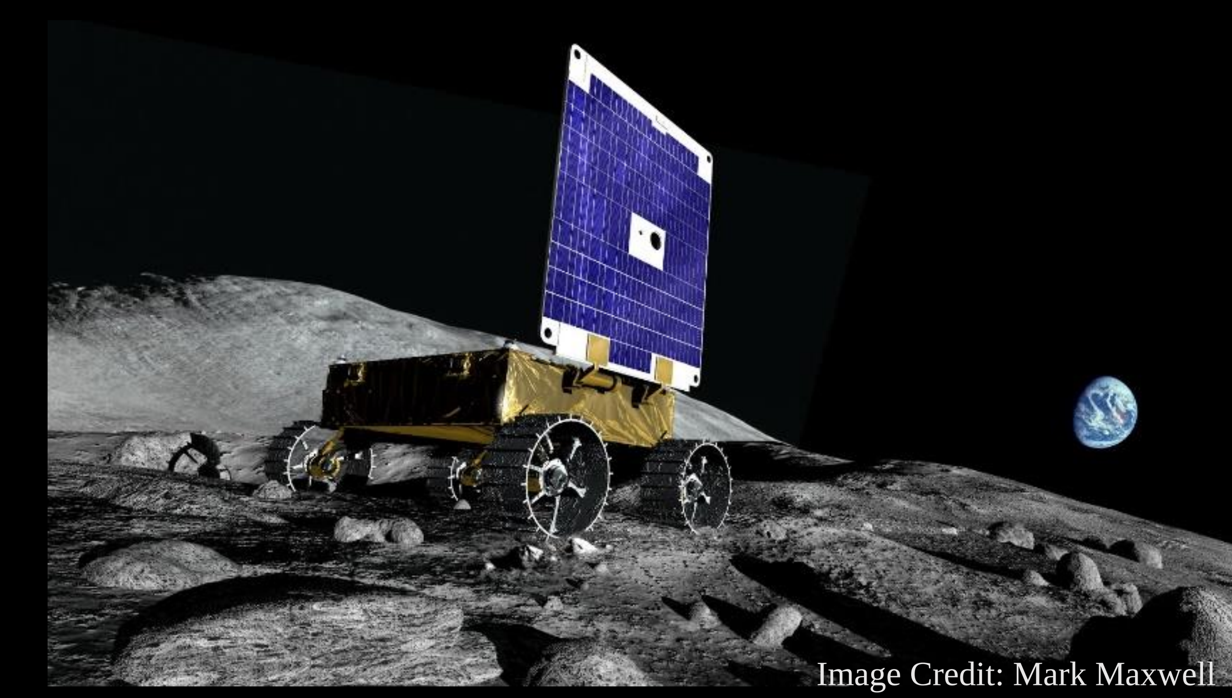


Image Credit: Mark Maxwell

IMPACT:

This work creates a complete, robust, and generalizable estimation procedure for robots to estimate their position in long treks on barren terrain. Computational constraints and perceptive challenges are alleviated by the usage of laser line striping and the continuing development of robust vision-denied pose estimation. The resource-limited exploration autonomy developed here will guide the first lunar microrover in search of polar ice, manifestation of this work in a world beyond.

NEXT STEPS:

Current pose estimation, especially for a small rover, drifts excessively with long distance. In the case of MoonRanger, which must return after long treks, the rover would be lost. Future measures intended in this research are to:
(1) Investigate alternative formulations of covariance matrices of pure encoder-based pose estimates and sensor readings to yield the most robust pose estimate using the Extended Kalman Filter.
(2) Customize inclusion of sun sensor data such that sun orientation, an absolute bearing available whenever the sun is visible, is averaged into the pose estimate, with consideration for how the sun would move over the course of an autonomous trek.
(3) Define constants in proposed vision-based algorithm, implement, and refine.
(4) Optimize state storage such that parameters are stored to facilitate continuation of pose estimation even if the state-dependent program was restarted via a fault, fault recovery, or power conservation procedures.

REFERENCES:

[1] "differential_drive – ROS Wiki." ROS Wiki, http://wiki.ros.org/differential_drive.
[2] "Gyroscope." Gyroscope, learn.sparkfun.com/tutorials/gyroscope/all.
[3] "How Do I Compute the Covariance Matrix for an Orientation Sensor? [Closed] Edit." ROS Answers, answers.ros.org/question/9446/how-do-i-compute-the-covariance-matrix-for-an-orientation-sensor/.
[4] "Lab 3: Line Following and Odometry." 16-311 Lab 3: Line Following and Odometry, Carnegie Mellon University , www.cs.cmu.edu/~16311/current/labs/lab03/index.html.
[5] "robot_pose_ekf – ROS Wiki." ROS Wiki, wiki.ros.org/robot_pose_ekf.
[6] "The Covariance Confusion." Athitya Kumar, athityakumar.github.io/blog/posts/The_Covariance_Confusion/.
[7] Carnegie Mellon University. "Downloads - The CMU Brand - Carnegie Mellon University." Downloads - The CMU Brand - Carnegie Mellon University, www.cmu.edu/brand/downloads/index.html.
[8] Gkioulekas, Ioannis. "Feature and Corner Detection." 16385: Computer Vision. 2020.
[9] Husky. "Husky/Husky." GitHub, Clearpath Husky, 20 Apr. 2020, github.com/husky/husky.
[10] Martinelli, Agostino. "MODELING AND ESTIMATING THE ODOMETRY ERROR OF A MOBILE ROBOT." InfoScience, infoscience.epfl.ch/record/97432/files/Martinelli_ifacnolcos.pdf.
[11] Schleppe, John B. "Development of a Real-Time Attitude System Using a Quaternion Parameterization and Non-Dedicated GPS Receivers ." University of Calgary , 1996.
[12] Uland Wong, Ara Nefian, Larry Edwards, Xavier Buoyssounouse, P. Michael Furlong, Matt Deans, and Terry Fong. Polar Optical Lunar Analog Reconstruction (POLAR) Stereo Dataset. NASA Ames Research Center, May 2017.
[13] Whittaker, Red. "Line Striping." 16865: Space Robotics. 2020.
[14] Xsens. "MTi 10-Series." Xsens, www.xsens.com/products/mti-10-series.

THANK YOU, Dr. Whittaker, for your practical mentorship! Thank you, Dr. Heather Jones, Dr. David Wettergreen, and Srini Vijayarangan for your guidance!