Real-time Algorithmic Detection of a Landing Site using Sensors aboard a Lunar Lander

Ina Fiterau
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
mfiterau@cs.cmu.edu

Venkat Raman Senapati
Electrical & Computer Engineering
Carnegie Mellon University
Pittsburgh, PA 15213
vsenapat@andrew.cmu.edu

Andrew Sheng
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
asheng@andrew.cmu.edu

Nagasrikanth Kallakuri
Electrical & Comp. Engg
Carnegie Mellon University
Pittsburgh, PA 15213
nkallaku@andrew.cmu.edu

Robert Walzer
Elec. & Comp. Engg
Carnegie Mellon University
Pittsburgh, PA 15213
ruw@andrew.cmu.edu

Abstract

Planetary Landers can achieve safe, pinpoint landings. This revolutionary capability is possible due to Hi-Def, full coverage photography for Moon & Mars. This paper presents a system, encompassing a suite of sensors and algorithms, capable of perceiving and interpreting information about the Lander's position relative to its surroundings - lunar coordinates, azimuth, altitude, attitude, velocity - as well as determining characteristics of the target environment - elevation and obstacles of the lunar surface. Most importantly, the paper describes how the information is used by path planning algorithms to determine areas on the surface where landing is feasible.

1 Introduction

Pinpoint landing is a critical contribution to any space mission with a distinct destination, for example, Landing within 100m is one of the contract items in the Innovative Lunar Demonstration Data (ILDD) with NASA. The final landing position is crucial, as any obstacles under the Lander when it touches ground should be avoided for safe landing. Moreover, the solar panels must be oriented such that the amount of captured solar energy is maximized. This paper shows how these objectives are achieved by computing robust estimates of the Lander's coordinates, attitude and trajectory as well as through sensing of the environment where it is expected to land.

Ultimately, the Lander uses this information to assess risks and pick a danger-free landing surface to set down securely.

In order to drive the Lander to a safe landing site the information required includes – velocity of the Lander, altitude from lunar surface, orientation with respect to the lunar surface, direction of gravity, azimuth and surface roughness estimation. A lab-analog sensor suite has been developed with sensors of small range to acquire the aforementioned types of information. Two Ground Speed sensors are considered as Doppler Radio Detection and Ranging (RADARs), a Hokuyo Laser range finder acquires three Dimensional LIDAR data, a Lucam industrial camera represents the camera on board. Tests have been conducted using these representative sensors to verify the algorithms. The specifications of the sensors were obtained from the tests. A physical model of the lunar surface was created modeling the pictures of lunar surface obtained from the Lunar Crater Observation and Sensing Satellite (LCROSS). The model has craters, obstacles, unevenness and lunar dust spread over it to simulate the actual lunar surface conditions. This helped in generating a test data set which was used for the verification of algorithms.

Optical flow algorithms were developed to be used as input for tasks like navigation & surface-feature tracking. The motion estimation vectors obtained by optical flow were used to determine surface velocity and time-to-contact. Lucas-Kanade method was used to solve the optical flow equations obtained by assuming neighboring pixels have the same intensity in a 4 X 4 pixels window. A single iteration of Lucas-Kanade method did not give good results hence multiple iterations were repeated to obtain better results. The motion estimation between frames was also obtained using the algorithm mentioned in [1].

Camera data from the National Aeronautics and Space Administration (NASA) LCROSS mission and Light Detection and Ranging (LIDAR), also Laser RADAR (LADAR) data from the NASA Lunar Reconnaissance Orbiter (LRO) mission formed one important dataset for training and testing the Markov Random Field (MRF) algorithm. LCROSS was a robotic lunar mission with the objective of exploring the material properties of a lunar crater; this was accomplished via the collision of a robotic probe with the lunar surface. On the other hand, LRO was a lunar mapping probe dedicated to studying lunar surface features from orbital height. These two missions were chosen due to the fact that LCROSS offered a high-resolution sequence of surface images from the viewpoint of a “landing” vehicle while LRO offered detailed LIDAR data of the region chosen by LCROSS. In order to form training datasets, LIDAR data points were mapped to LCROSS images based on pose data derived from information provided with the images.

One of the main contributions of this paper is the use of terrain characteristics to decide on a landing spot that not only makes it easy for the landing stage, but also ensures that the rover can egress and move toward the objective of at least move 500m in a direction. The features used was terrain elevation - more specifically, slope as the gradient of elevation. Limitation of the Lander structure and the rover's motion were used to obtain the map of obstacles needed by the planning algorithm. Candidate landing places were obtained by use of a moving window scan across the landscape. The planning algorithm checks the feasibility of each landing location candidate until one with a feasible solution for the rover is found.

To estimate terrain elevation Markov Random Fields were used to combine LIDAR readings with imagery and update the elevation estimation through coordinate descent. The basic principle used was that two pixels with a low distance in color value should also correspond to two altitude values that are not too different. Improvements were made to the algorithm to adapt it to the lunar landscape conditions, such as lowering the interaction between shaded pixels since the shaded regions of the landscape introduce uncertainty. In testing the MRF method against the LIDAR data from the LRO and lunar imagery from the LCROSS, it was concluded that the algorithm decreases the error from 16% - obtained through interpolation - to 12% after coordinate descent. The algorithm that chooses the landing spot was tested with both the orbital data and the small scale model data to a successful conclusion.

2 Sensors on board for the Lander

Lander sensors include three Doppler RADARs for velocity measurement, a set of cameras positioned at different places on the Lander, a laser altimeter, a flash LIDAR for obtaining the depth map of the surface of moon, and an inertial measurement unit. These sensors are used with specific settings and in different combinations to obtain the complete Lander state information. A test set up was made using representatives for the aforementioned sensors and a lunar surface model.

The representative sensor suite on which different algorithms and approaches are tested consists of a Laser Measurement System (LMS) as a Flash LIDAR, a Hokuyo scanning range finder URG laser as an altimeter, LuCam industrial camera as the camera. The methods by which the Lander state information can be obtained using these sensors is given in the following sections.

3 Lander state parameters

For the Lander to fly safely to the landing spot, the knowledge of its velocity, altitude and orientation are crucial. Any inaccuracy in measuring the mentioned state parameters leads to unwanted results in the Lander behavior. So the sensor suite has to be organized in the most useful way to obtain the required information.

3.1 Velocity measurement

The velocity information required is a three dimensional velocity vector of the Lander. Velocity of motion of the Lander in the entire x, y and z axes is required. This is realized by means of Doppler Radio Detection and Ranging (RADAR) sensors that give the velocity using the Doppler Effect. A Doppler RADAR is specialized RADAR that makes use of the Doppler effect to give the velocity data about objects at a distance. It does this by beaming a high frequency signal towards a desired target and capturing the reflection, then analyzing how the frequency of the returned signal has been altered by the object's motion. The variation in the frequency between the transmitted and received waves gives accurate measurements of the radial component of target's velocity relative to RADAR.

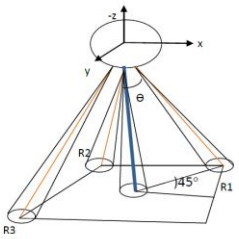
In our case, the target is the lunar surface. When the Doppler RADAR transmits high frequency signals on to the lunar surface and obtains the reflected wave, the velocity of the Lander is obtained from the difference in frequency as shown below. The exact result derived with c as the speed of light and v as the target velocity gives the shifted frequency (F_r) as a function of the original frequency (F_t):

$$Fr = Ft\left(\frac{1+\frac{v}{c}}{1-\frac{v}{c}}\right) \qquad Fd = Fr - Ft = 2v\frac{Ft}{c-v}$$

Since in most cases $v \ll c, (c - v) \to c$ $\qquad Fd \approx 2v\frac{Ft}{c}$

The Doppler RADAR gives the radial velocity of the Lander. In order to obtain the Lander velocity in three dimensions, a specific setup for the placement of the RADARs should be followed and the calculation of the velocity components in the *x*, *y* and *z* directions relative to the Lander can be done using the setup.

As there are three components of the velocity to be found, three Doppler RADARs are required. These are placed pointing at the three corners of a square on the target as shown in the figure. Each of these RADARs subtends a fixed angle of 30 degrees from the center of the Lander. The RADARs individually give a radial velocity vector of the Lander. These velocities can be divided into the *x*, *y* and *z* directed velocity vectors by solving for *Vx*, *Vy* and *Vz* in the following equations.



Figdure.1 Doppler radar setup on Lander

$$V1 = \frac{c\Delta f1}{fo} = \frac{VxSin\theta}{\sqrt{2}} + \frac{VySin\theta}{\sqrt{2}} + VzCos\theta \qquad V2 = \frac{c\Delta f2}{fo} = -\frac{VxSin\theta}{\sqrt{2}} + \frac{VySin\theta}{\sqrt{2}} + VzCos\theta$$

$$V3 = \frac{c\Delta f3}{fo} = -\frac{VxSin\theta}{\sqrt{2}} - \frac{VySin\theta}{\sqrt{2}} + VzCos\theta$$

From the above equations Vx, Vy and Vz can be solved as shown below.

$$Vx = \frac{V1-V2}{\sqrt{2}Sin\theta} \qquad Vy = \frac{V2-V3}{\sqrt{2}Sin\theta} \qquad Vz = \frac{V1+V3}{2Cos\theta}$$

Thus the Doppler RADARs can be set up to obtain the velocity data of the Lander during flight. This method yields almost accurate velocity of the Lander, depending on the precision of Doppler RADARs.

3.2 Orientation of the Lander

An on board Inertial Measurement Unit (IMU) gives the relative orientation of the Lander. The main problem with the IMU is that it accumulates errors gradually over time. It cannot be relied on for the exact orientation information. One more data source, like a Kalman filter, can be used with IMU to obtain more accurate orientation information. Tracking the sun or earth or stars can be done to have the complementing orientation information source. The main purpose of the Kalman filter is to use measurements observed over time, containing noise (random variations) and other inaccuracies, and produce values that tend to be closer to the true values of the measurements and their associated calculated values.

The sun or earth tracker is a high field of view camera, which sometimes has the ability to distinguish bright spots in the image frame. These frames are filtered to obtain the Earth or Sun and then find the difference in angle of the sun or earth from a reference position of the Lander, which gives the orientation of Lander. This value is used along with the data from the Kalman filter to obtain more accurate orientation of the Lander. Also, star tracking can be done to get additional information about the

attitude of the Lander. A set of stars and compare the image with the standard database and identify the attitude. But, there are some tradeoffs for this method since sometimes the stars become invisible due to the brightness of the sun or earth.

3.3 Altitude of Lander

The altitude of the Lander is obtained using a Laser altimeter on board. A Laser altimeter transmits a very high frequency (of the order of hundreds of thousands of hertz) radio waves on to the surface of the moon and counts the time till it receives back the wave. If *R* is the altitude of the Lander from the lunar surface, Δt is the time taken for the transmitted wave to reflect back and 'c' is the velocity of light, the altitude can be obtained from the equation, $\Delta t = 2R/c$.

3.4 Direction of Moon's gravity

The direction of the moon's gravity is very important to inform the Lander for safe landing with appropriate attitude. This can be obtained by again tracking the sun and earth and identifying the moon's surface. With the help of the IMU and position of sun and earth the approximate direction of moon's gravity is obtained.

3.5 Slope of the ground

The LIDAR is used in determining the slope of the ground. The flash LIDAR gives the elevation information as stripes of 2D range data. The slope of the ground can be obtained using any two such stripes of LIDAR 2d data and calculating the angle of elevation at the target surface of the LIDAR and at finding the difference between them as shown.

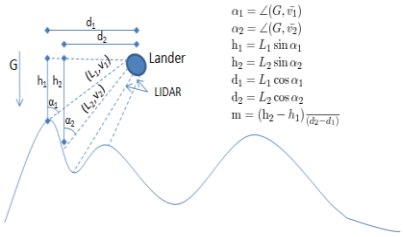


Figure 2 obtaining the slope of the lunar surface underneath the Lander using the LIDAR data.

4 Terrain Evaluation

Of equal importance to observing the lander state is keeping track of the terrain that the lander is descending towards. This serves the purpose of timely detection of hazards during the descent stage as well as validation of a safe landing spot. This paper deals with two aspects of terrain evaluation: modeling the lunar terrain and assessing the possibility of landing considering the characteristics of the lander. This section discusses how a Digital Elevation Map (DEM) can be obtained by combining sensor readings - more precisely, LIDAR and camera - and how this DEM can be put to use during the landing area selection process.

4.1 Elevation Estimation

Elevation estimation is a crucial aspect of the descent without which the craft could easily crash, land in a region that does not allow rover egress or movement. One approach to this task is to obtain a DEM based on the coordinates and orientation of the lander through the transformation of the corresponding segment - depending on the lander location - of a stored elevation map of the moon. While this only requires a rotation - depending on the orientation of the lander - there is the overhead introduced by the requirement of storing an altitude map of the entire moon - or at least a significant portion of it. Additionally, there the main risk associated with this method is that if the coordinates are wrong or the map is not accurate the DEM will also be wrong with devastating results.

A more prudent but potentially costly approach involves building an elevation map based on instantaneous sensor readings. This requires intense computation and can go wrong if the readings are wrong, but otherwise provides a DEM of the actual terrain that the lander is located above. LIDAR readings alone are insufficient to obtain a DEM because of their scarcity. Also, images offer some cues as to the terrain, however, exact altitudes cannot be inferred as techniques such as shape from shading overestimates the steepness of the landscape. Nevertheless, the two can be combined through the use of a Markov Random Field (MRF) [2]. The MRF is a graphical model that looks like a grid and in which each variable is connected to at most four neighbors: top, bottom, left and right. The connection between two neighbors is given by the distance between pixel values in the image. Also, the existing LIDAR measurements are connected to the corresponding nodes in the rid. In the equations below, the notations used are X for the image matrix, Z for the sparse elevation measurements, Y for the estimated elevation map (the grid), L to denominate the points where elevation readings exist, $N(i)$ to denote the neighborhood of point i on the grid and w_{ij} to represent the correlation between pixels in an image. The equation below represents the objective function for the MRF.

$$argmax_y \exp(-\frac{1}{2}(\sum_{i \in L} k(y_i - z_i)^2) + \sum_i \sum_{j \in N(i)} w_{ij}(y_i - y_j)^2))$$

The MRF requires perfect alignment between the LIDAR readings and the image - it is imperative that the corresponding pixel for each LIDAR reading is known. Also, the model needs a starting point that is not completely removed from reality. For instance, marking the unknown readings with 0 will heavily and negatively impact the accuracy of the estimation, since the proportion of existing versus non-existing readings is typically 100:1. In this case, the LIDAR data is interpolated by initializing each point with a weighted average of the nearest neighbors on each direction. The weights used are inversely proportional to the distance.

It is notable in the images used, that the pixels in the shaded regions do not necessarily denote close values of elevations; it might simply be that the landscape features are obscured by the shadow. In such instances, it would be preferable if the image influenced the update process in the least amount possible. The shading coefficient is introduced to this purpose. The shading coefficient makes it such that the points darker than a threshold have very little influence on the update process. The benefits and overhead introduced by real-time elevation estimation are discussed in section 5.

4.2 Landing Zone Selection

During the final portion of the descent, a landing spot must be selected. Although the coordinates of the landing site are generally known, the lander must be able to select its landing spot in real time in view of unforeseen circumstances such as it being located over a different segment of the surface than anticipated. The landing zone selection algorithm processes the DEM to obtain gradients at each point and slopes between two points in the landscape. The lander features used in making the decision are the maximum supported difference in elevation that the legs can withstand. Large portions of the terrain can be disregarded from consideration based on the steepness. Other areas are too curved - although the gradient is close to 0, the area is either below or above surrounding landscape. In order to select candidate landing spots from the remaining regions, the terrain is scanned with a window of the size of the lander and all feasible landing areas are marked - feasible means that the legs are not strained, there is sufficient clearance for the lander body and sufficient spacing from hazardous terrain makes egress possible.

Subsequent to selecting a list of potential landing zones, a process called look-ahead path planning is initiated. The path planning algorithm is needed to ensure that the rover is capable of moving along a given distance. A pre-step to the spatial planning eliminates regions of the landscape that are impracticable because of the steepness. The parameters used in this elimination are the pitch, roll and yaw constraints of the rover - meaning what are the maximal angles that the rover can withstand. After the infeasible regions are eliminated, the process is reduced to a classical 2D planning algorithm with a rectangular agent. The testing section illustrates examples of the performance of this algorithm.

5 Testing

This section describes the tests were performed using the terrain evaluation on data available from NASA - from the LRO and the LCROSS mission - and on data obtained by test sensors on a miniature landscape made to mimic the lunar surface. Two types of tests were performed, one to test the importance of the weighting parameters between the LIDAR and the readings and the second to illustrate the performance of MRF during descent. Each of these two types of tests was run with the Cabeus and the surface model data. Appendices D1 and D2 present the results of these tests.

In the parameter test, the weight the image data has in the update process is gradually increased. Variation of error with parameters shows that the image is as useful as LIDAR readings in reducing error. The tests on the surface model show that sensor tolerance is low enough to ensure reduction of error. Also the lighting conditions have a considerable effect on the performance

Overall, the algorithm obtains a 3-10% reduction of error, with the error mostly distributed where the terrain gradient is higher. Evaluation of runtimes on a quad core with 16 Gb RAM have shown that at least one MRF run can be executed in real time.

6 Conclusion

With the combination of camera, LIDAR, RADAR, and IMU, a lunar lander can achieve pinpoint landing. Using these sensors velocity, altitude, and orientation can be determined. In addition, surface roughness and slope can be estimated to allow for accurate path planning and landing zone selection. Each of these sensors alone would not be able to accomplish these tasks, but with the developed algorithms the sensors can now be utilized more effectively to accomplish the goals.

Up to this point, these algorithms have been run on servers with more computational power than would be available during a mission. In order to achieve accurate landing, they will need to be run in real time on the lander. The path planning algorithm will thus need to be optimized for computational efficiency. Optical registering, which uses the relative position of known points to determine the position of the lander, can be implemented to increase the robustness of this sensor package. Stereo vision for the lander will also enable the lander to utilize more accurate visual odometry algorithms.

References

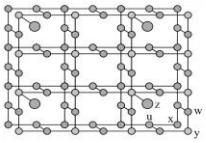
[1] Howard, A., "Real-time stereo visual odometry for autonomous ground vehicles," Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on, vol., no., pp.3946-3952, 22-26 Sept. 2008

[2] Diebel J. and Thrun S.;, “An Application of Markov Random Fields to Range Sensing”, Proceedings of Conference on Neural Information Processing Systems, 2005

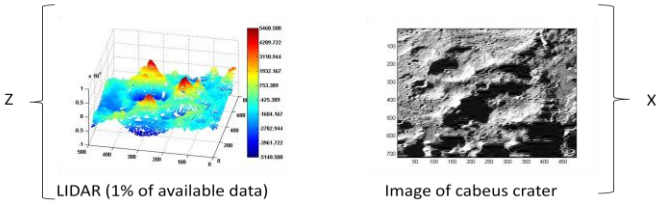
[3] R.K.Cheng – “Surveyor Terminal Guidance”

Appendix A: Markov Random Field applied to LRO LIDAR with LCROSS imagery

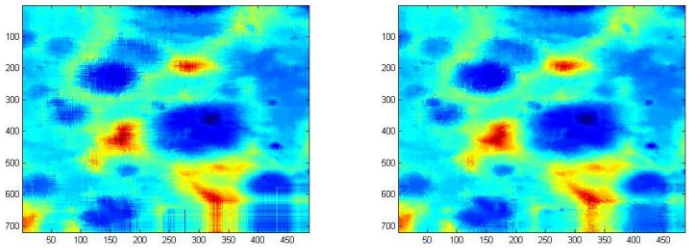
X – image
Z – sparse elevation measurements
Y – estimated elevation map
L – points where elevation readings exist
N(i) – neighborhood of point i on the grid
w_{ij} – correlation between pixels in image



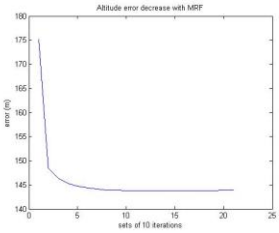
$$argmax_y \exp(-\frac{1}{2}(\sum_{i \in L} k(y_i - z_i)^2) + \sum_i \sum_{j \in N(i)} w_{ij}(y_i - y_j)^2))$$



Overview of the MRF algorithm



Elevation map after interpolation and after 200 iterations of coordinate descent



Decrease of average elevation estimation after applying MRF from 16% to 13%

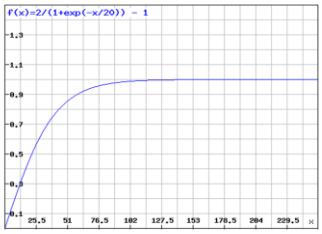
Pixel correlation

$$w_{ij} = q \times exp(-c||x_i - x_j||)$$

Shading coefficient

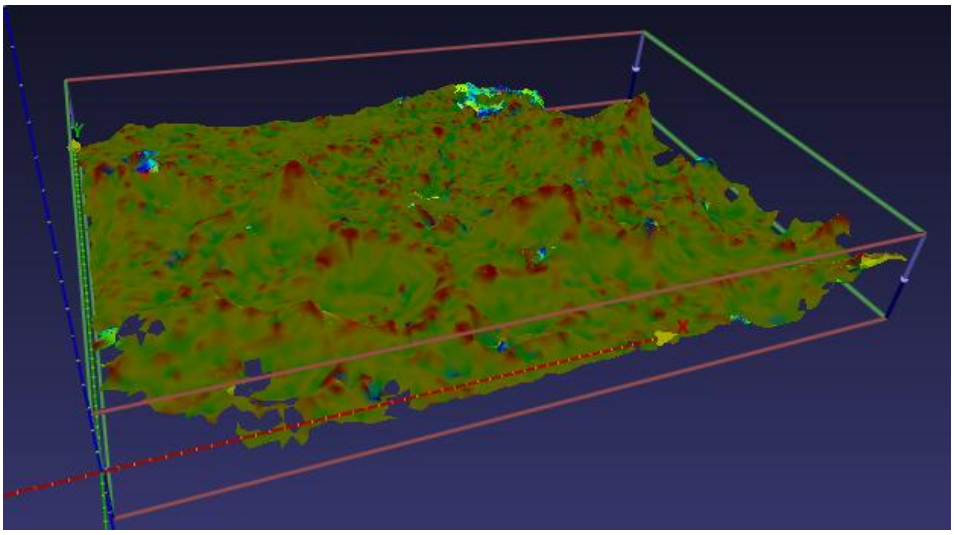
$$q = \frac{2}{(1 + \exp(-\frac{x}{p}))} - 1$$

p =5% quantile of image data
– the shaded pixels

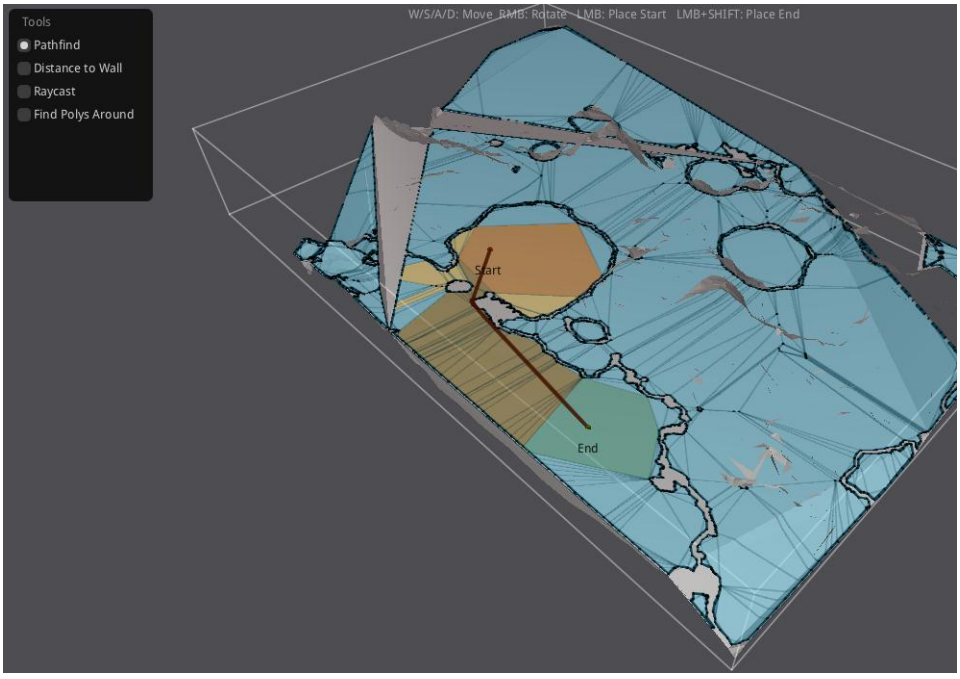


Shading coefficient - formula and plot against the pixel color

Appendix B: Example of Planning Map with rover approximated by a point

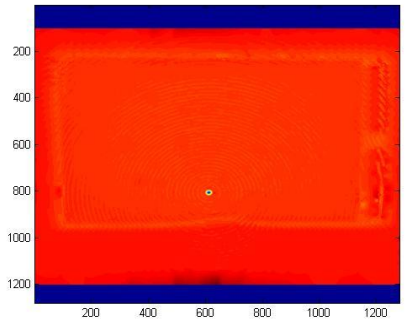


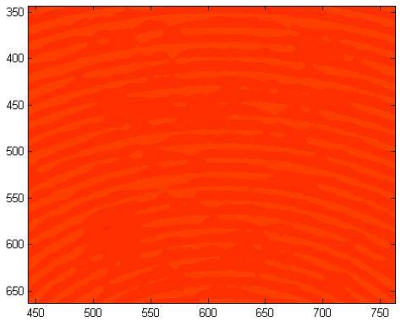
Gradient map of the landscape around the Cabeus crater

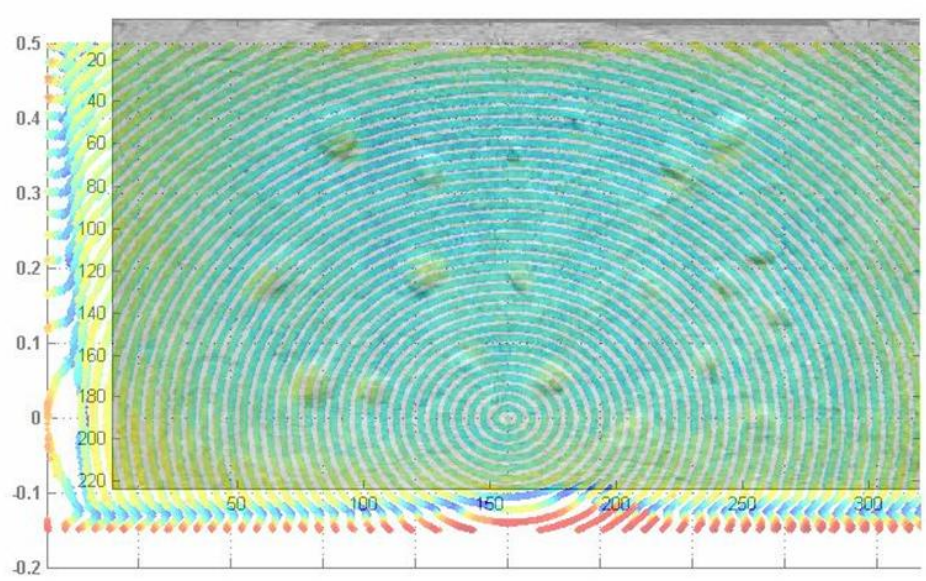


Path planning based on the gradient map

Appendix C: LIDAR readings from the Laser measurement System of Lunar surface model for testing. LIDAR readings overlapped with camera capture.







Appendix D1: Results of parameter variation test

Surface model

Resolution	325x500
LIDAR readings	391
Base Error	0.94

Image Weight	LIDAR Weight	1Iteration error (cm)	1Iteration runtime (ms)
0	1	0.8313	46.89
0.1	0.9	0.8349	47.82
0.2	0.8	0.8362	46.59
0.3	0.7	0.8368	45.39
0.4	0.6	0.8372	45.57
0.5	0.5	0.8375	45.67
0.6	0.4	0.8377	46.71
0.7	0.3	0.8378	46.84
0.8	0.2	0.8380	46.47
0.9	0.1	0.8380	46.84
1	0	0.8381	46.51

Cabeus Data

Resolution	120x180
LIDAR readings	1124
Base Error	64.5

Image Weight	LIDAR Weight	1Iteration error (m)	1Iteration runtime (ms)
0	1	58.5765	56.6194
0.1	0.9	58.5220	57.1578
0.2	0.8	58.5129	57.1102
0.3	0.7	58.5098	57.2784
0.4	0.6	58.5085	56.7692
0.5	0.5	58.5075	56.6982
0.6	0.4	58.5072	56.9431
0.7	0.3	58.5072	56.6624
0.8	0.2	58.5086	57.3292
0.9	0.1	58.5085	56.9162
1	0	58.5069	57.0678

Appendix D2: Results of Descent Simulation by zoom-in

Surface model

Height (m)	Error (cm)	No. of LIDAR Readings	Runtime (s)
1.50	1.0398	552	0.2804
1.39	0.6356	668	0.2754
1.28	0.6566	770	0.2785
1.16	0.6186	945	0.2775
1.05	0.6746	1165	0.2769
0.94	0.6036	1508	0.2816
0.83	0.5917	1888	0.2687
0.71	0.5727	2445	0.2712
0.60	0.4401	3052	0.2942
0.49	0.4434	3477	0.2787
0.38	0.6081	3910	0.2646

Cabeus Data

Height Scale	Error (m)	No. of LIDAR Readings	Runtime (s)
1.00	98.4025	668	0.7243
0.93	96.7410	739	0.7255
0.85	67.2593	810	0.7280
0.78	60.6598	906	0.7098
0.70	55.1811	999	0.7106
0.63	51.1427	1125	0.7069
0.55	48.7305	1256	0.7082
0.48	45.9367	1350	0.7106
0.40	42.9536	1350	0.7159
0.33	42.0494	1350	0.7035
0.25	38.3335	1350	0.7030