A Landmark Based Position Estimation for Pinpoint Landing on Mars

Yang Cheng and Adnan Ansar
Mobility Systems Concept Development Section
Jet Propulsion Laboratory, California Institute of Technology
4800 Oak Grove Drive, Pasadena, CA 91109-8099, USA
ycheng@telerobotics.jpl.nasa.gov

Abstract *- Real-time position estimation for a descent lander is a critical technological need for many of NASA's planned in situ missions including landing on a number of bodies at locations of greatest scientific interest and sample return. In particular, it enables the capability to land precisely and safely in a scientifically promising but hazardous site and is a key technology to be demonstrated by NASA in the next decade. The key challenge of pinpoint landing (PPL) is how to localize the lander by recognizing the landmarks (craters) in the landing area and match them positively to a preexisting landmark database while the spacecraft is descending. In this paper, a real-time landmark based position estimation technique for pinpoint landing is suggested. This system includes three crucial components: (1) real-time landmark detection, (2) real-time landmark matching and (3) state (both position and velocity) estimation. We discuss the performance analysis of this system. Finally, we show that the suggested technology is able to deliver a spacecraft to less than 100 m from a pre-selected landing site on Mars.*

Index Terms: *Pinpoint landing, landmark detection, landmark matching, and position estimation. In situ missions*

I. INTRODUCTION

Current descent and landing technology for planetary missions, such as landing on Mars, is characterized by at least a 30 x 100 km landing error ellipse, with no terrain recognition or hazard avoidance technology utilized to date. In the next decade, NASA will demonstrate a very ambitious capability -- pinpoint landing to other planetary bodies. The primary objective of PPL is to deliver a spacecraft to a targeted landing site within less than 100 meters. In order to accomplish this, several new technologies will be developed in the next few years, including new optical guided navigation system for precision entry, optimized powered descent guidance system, advanced parachuting technology, a new propulsive traverse capability and a new real-time terrain recognition for spacecraft localization capability.

Automated landing systems have been flown successfully by NASA Lunar and Mars exploration missions such as *Surveyor, Apollo Lunar Module, Viking, Mars Pathfinder* and *Mars Exploration Rover (MER)* missions. However, none of these missions except one had a capability for terrain-relative guidance, navigation and control (GNC). In MER, a simple Descent Image Motion Estimation System (DIMES) was used in estimating horizontal velocity. DIMES consisted of a descent imager, a radar altimeter, an inertial measurement unit and an algorithm to provide a low cost, robust and computationally efficient solution to the horizontal velocity estimation problem, which is critical for the safety of the air bags during the touch down [9]. DIMES is the first ever terrain-relative sensing and guidance system used by real mission. However, DIMES does not have either terrain recognition or hazard avoidance capabilities. Therefore, it does not meet the requirements of a PPL mission.

The real-time spacecraft localization scheme works as follows. First, a targeted landing site on the targeted body is selected on the earth using orbital imagery, and the landmarks (e.g. craters) within the landing ellipse are mapped. During the lander descent, its initial position with respect to the landmarks as well as to the selected landing site is determined automatically on board. The lander is then guided to the landing site using continuous updates of lander position and velocity throughout the descent. Three fundamental requirements for this new technology are:

1. The system must be able to recognize the terrain (landmarks) reliably and repeatedly in a highly variable environment. During the spacecraft descent, considerable environmental variations, such as lighting, atmospheric conditions, viewing angle, spacecraft altitude, etc., could affect the appearances of landmarks. The key to success relies on defining or choosing a class of landmarks which have good invariance properties under variable environmental conditions and a set of associated algorithms for handling this type of landmark robustly.
2. The system must accomplish the task under extreme conditions imposed by the slow flight computer and terminal descent time constraints. The projected CPU clock speed of the flight computer for a PPL mission will be on the order of 100MIPS, which is not ideal for processing large volumes of data. For example, a descent image is typically 1 MB in size. There is a roughly 60 second window of opportunity for spacecraft localization during the parachuting stage between the heat shield jettison and powered descent. In order to obtain reliable spacecraft state information (position and velocity) several images as well as other onboard sensor data such as IMU and altimeter data have to be processed during this period of time. In general, each cycle of spacecraft

localization, which includes image acquisition, image processing and sensor data infusion, should be done in a few seconds. This is a very demanding task.

3. The system must be able to guide the spacecraft to land within 100 meters of the target under noisy conditions. Due to the limitations of both hardware and software, system noise can only be compensated to a limited extent. The primary noise sources are the landmark detection error, the base map error (both position and elevation), sensor noises (imager, IMU, altimeter), and image and IMU sensor misalignment. All of these influence the performance of the system. By considering these uncertainties, the system must be robust enough to provide valid spacecraft state, which meets or exceeds the PPL requirement.

The rest of this paper, we will present a system, which can meet these requirements.

II. SYSTEM DESCRIPTION

Craters are landforms commonly found on the surface of planets, satellites, asteroids, and other solar system bodies. A crater, in general, is a bowl shaped depression created by collision or volcanic activities. Because of their simple and unique geometry and relatively stable appearance under different viewing and lighting angles, craters are ideal landmarks for spacecraft localization [1-3]. A large number of craters can be found on the surface of Mars, which can generally be divided into three regions – heavily, moderately and lightly cratered areas [5, 10]. A statistical study shows that there will be adequate craters in a landing ellipse to ensure positive spacecraft localization.

We suggest craters as landmark for the system, which includes four components: real-time crater detection, crater matching, space craft position estimation, and velocity estimation.

Real-Time Crater Detection

A real–time crater detection algorithm has been developed based on a previous version of the crater detection algorithm for autonomous spacecraft navigation [2]. The crater detection algorithm takes five steps to do the detection:

1. Edge Detection: This step detects edges in an image and places them in a database.
2. Rim Edge Grouping: This step groups together edges that belong to the same crater. The information used for this process includes edge shape (convex), the image intensity profile inside a crater, edge gradients. If a pair of edges (lit and shaded side of crater) is found, they will be used to fit an ellipse.
3. Ellipse Fitting: This step fits an ellipse to each group of crater edges by an iterative algorithm – the reweighting least squares method, which could robustly remove any outliers in the input points.
4. Precision Fitting: This step adjusts the detected crater's geometry directly in the image domain to reduce errors introduced in edge detection and ellipse fitting. A multidimensional iterative nonlinear minimization algorithm based on conjugate gradients is used to lock an ellipse precisely onto the rim of a crater.
5. Crater Confidence Evaluation: This step evaluates every detected crater and assigns a confidence value to it.

After extensive optimization, this algorithm is able to meets the PPL time and performance requirement for the first time. The improvement of this algorithm is due to the following changes

1. Reorganization to allow maximum usage of pointer operations.
2. Conversion and analysis of image features in vector data format.
3. Use of a hash table vector database for fast data retrieval.
4. Extensive use of lookup tables for repetitive computations.

The current algorithm is able to detect craters from a 512 x 512 image in less than 0.8 second on a 333 MIPS Ultra SUN station, which meets the Mars PPL requirement.

.

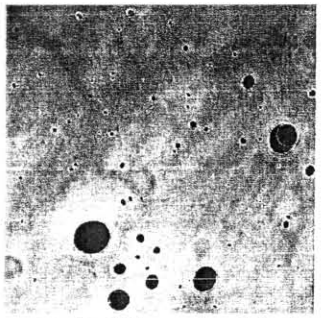

Figure 1: An example of the crater detection result from MER–A descent imagery. The detected craters are shaded for easy display purpose.

Extensive experimental studies show that the detection rate is better than 94% and the false alarm rate is less than 7%. The position error is less than 0.3 pixel and the geometrical error is less than 0.5 pixel.

Crater Matching

In order to determine the position of the spacecraft with respect to the central body, geometrical recognition techniques that perform matching between the craters extracted from an image and a crater database containing the 3D locations of the craters are used. Each crater is treated as an attributed point corresponding to the center of the crater, where the attributes are the radius and

orientation of the crater. The efficiency of the basic methodology is improved by two means. First, the crater attributes are used to remove matches that are incompatible. Second, an initial estimate of the spacecraft position is used to filter matches that are not feasible. The information from other sensors such as the altitude from the altimeter and attitude from the IMU can help to reduce the search scope even further. Conic invariants provide another useful criterion [4]. A pair of coplanar conics c_1 and c_2 has two invariants

$$I_{c1c2} = Trace(c_1^{-1}c_2)$$
$$I_{c2c1} = Trace(c_2^{-1}c_1)$$

Since under a linear transformation $\boldsymbol{x} = \boldsymbol{TX}$, c_1 and c_2 go to $\boldsymbol{C}_1 = \boldsymbol{T}^t c_1 \boldsymbol{T}$, and $\boldsymbol{C}_2 = \boldsymbol{T}^t c_2 \boldsymbol{T}$, we have

$$I_{C1C2} = Trace(T^{-1}c_1^{-1}(T^t)^{-1}T^t c_2 T)$$
$$= Trace(c_1^{-1}c_2) = I_{c1c2}$$

The same derivation holds for I_{c2c1}.

A Mars Orbiter Camera (MOC) image strip, which is 2048 by 8366 in size with approximate 1.41 meter pixel resolution, was used as the base map. A total of 1,777 craters were detected from it. The crater matching algorithm has successfully matched the craters from Fig. 1 to this map. Fig. 2 shows the results in a subset of the MOC image.

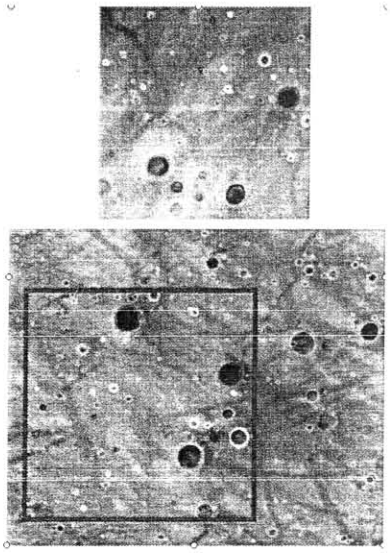

Figure 2: An example of the crater matching result between MER descent image (top) and MOC image. The color rim indicates the correspondences.

Robust Position Estimation

From a suitable number of matched landmarks for which we have prior geometric data, we accurately estimate the position and orientation of the spacecraft with respect to the surface of the planetary body. In the case of craters, the relevant structures are crater centroids, which are estimated carefully to account for perspective distortion effects.

Given a collection of points in 3D and their 2D projections, we recover camera pose as follows. The 3D points are originally presented with respect to some reference coordinate frame, typically dependent on the landing ellipse and independent of the location of the camera. The first step in recovering camera pose is to determine the coordinates of these points in a coordinate frame centered on the camera. From prior calibration of the camera, we know the exact 2D coordinates of a pixel on the image plane (CCD or CMOS device). If (x,y) are the 2D coordinates of an image point p arising from a 3D point P, then P can be expresses in the coordinate frame of the camera as $(\lambda x, \lambda y, \lambda)$ for some suitable scale factor λ. Note that the distances between 3D points are independent of coordinate system. Hence, for a collection of image points $\{p_i\}$ and associated 3D points $\{P_i\}$, we know $\{d_{ij}=\|P_i-P_j\|\}$. This can be expressed as

$$d_{ij}^2 = (\lambda_i x_i - \lambda_j x_j)^2 + (\lambda_i y_i - \lambda_j y_j)^2 + (\lambda_i z_i - \lambda_j z_j)^2$$

resulting in a set of quadratic equations in the unknown $\{\lambda_i\}$. We use an efficient and robust linear algorithm to solve for the λ_i [6]. Once these quantities are known, the 3D coordinates of all points are known in both the coordinate frame of the camera and the reference frame. Recovery of the camera pose is then equivalent to finding the Euclidean transformation, which maps one of these point clouds onto the other. This absolute orientation problem has several known linear solutions [6]. For small numbers of points, the proposed method is fast and robust. It requires no initialization because there is no iterative component. Consequently, there is no convergence or local minima issues. Given an initial guess, we use a fast iterative method [8] to refine the result. Finally, we use a robust estimation approach to decrease sensitivity to outliers. We perform position estimation over a statistically meaningful number of trials using a subset of the data. The Least Median Squares of reprojection errors is used as the evaluation criterion. We also use early termination if the maximum error is small enough.

III. PERFORMANCE ANALYSIS

The ultimate objective of a pinpoint landing is to deliver a spacecraft within 100 meters from the targeted site. However, the uncertainties of hardware, software and landing site topography can have influences on the system performance. The primary noise sources are the landmark detection error, the base map error (both position and elevation), sensor noises (imager, IMU, altimeter), and image and IMU sensor misalignment. Although these uncertainties cannot be eliminated, we hope that their influence will be within the landing error envelope. In addition, the terrain recognition, the crater matching, must be unambiguous. In this section, we will study the influence of a few major uncertainties.

Crater Constellation Uniqueness Analysis

To be used for unambiguous position estimation, the constellation of landmarks must be unique in terms of size and location in the landing ellipse. We have conducted two studies in this area. The first study used an Odyssey THEMIS image, which contains 917 detected craters (Fig. 3). A set of neighboring craters forms a crater constellation and we compare it with other constellations in terms of their size and relative configuration (to within 1 pixel).

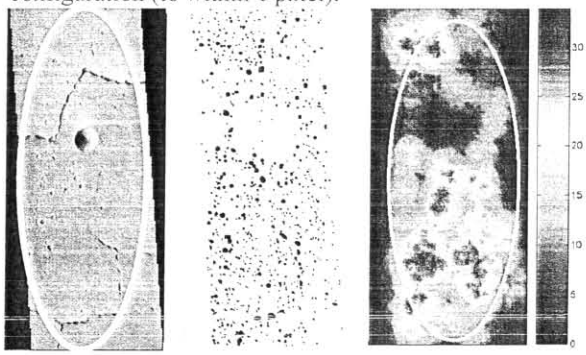

Figure 3: Odyssey THEMIS image (left). Detected craters (center). Crater density map (right)

The probability of ambiguous configuration in this dataset is shown in Table 1. This indicates that given the size and position of craters, the probability of ambiguous configurations is very small (< 0.0001%) when the number of craters in a constellation is greater than 5.

Table 1: Probability of ambiguous configuration from a real crater database

Number of craters	Total # valid configuration.	# Ambiguous configuration	% probability of confusion
2	7.1×10^5	71,898	10.1
3	1.3×10^7	7,394	5.8×10^{-2}
4	1.8×10^8	1027	5.7×10^{-4}
5	2.2×10^9	221	1.0×10^{-5}

The second study is a statistical analysis of the likelihood of confusion given a model for crater distribution.[10] This again takes into account the crater sizes and relative distances. We present a sketch of the derivation and the results. The probability of having two pairs of craters at the same relative distance d in a disk (region of attention) of radius R can be computed as:

$$P_1 = \left(\frac{3}{4}\frac{1}{R}\arctan\left(\sqrt{\frac{4R^2}{d^2}-1}\right)\right) - \left(\left(\frac{d}{4}\right)^2 \frac{3}{R^3}\sqrt{\frac{4R^2}{d^2}-1}\right)$$

We omit the derivation of this and some other quantities for space considerations.

Assuming a tolerance δ_p for distances to be considered "equal," the probability of having any two pairs of craters at the same relative distance in the disk is given by integrating P_1 over all admissible distances to get a probability P_2. Suppose there are N total craters in the disk and we use m for position estimation, leaving $n=N-m$ remaining. The probability of having another m craters with the same geometric configuration can be computed as

$$P_{geom} = (1-(1-\frac{\delta_p^2}{R^2})^n)^{m-2}\left(1-\left(1-P_2\right)^{(2n-3)(2m-3)}\right)$$

Given a model for crater distribution from [10], we find by integrating over all craters from size d_{min} to d_{max} that the total number of craters in our disk is:

$$N = A \cdot \frac{K}{1-\alpha}\left(d_{\max}{}^{1-\alpha} - d_{\min}{}^{1-\alpha}\right)$$

where A is the area of the disk and K and α determine crater size distribution. This also gives a probability density $\sigma_N(d_{min},d_{max}) = N/A$ for craters within the given size. The probability of having two craters with the same size (up to a tolerance δ_s can by computed by integrating $\sigma_N(d,d+\delta_s)^2$ over all admissible d. Let this quantity be P3. Then the probability of m craters out of a candidate population of n having the same size as our m selected craters is

$$P_{size} = (1-(1-p_3)^n)^m$$

Finally, the probability of an ambiguous constellation is the product of the size probability and the geometric probability

$$P_{tot} = P_{size}P_{geom}$$

For a heavily cratered region, we get $(K,\alpha) = (0.3, 1.8)$ from [10]. With disk diameter = 8 km, $\delta_s = \delta_p = 30$ m, $d_{max} = 4$ km and $d_{min} = 100$ m, we obtain the results in Table 2. These match the results in Table 1 up to order of magnitude, which is as much as we can expect for this relatively vague notion of "heavy" cratering..

Table 2: Probability of ambiguous configuration from crater distribution models.

Number of craters	Lightly cratered	Heavily cratered
2	5.4	44.4
3	5.2×10^{-3}	0.20
4	2.9×10^{-6}	8.4×10^{-4}
5	1.4×10^{-9}	3.4×10^{-6}

Position Estimation Analysis

From a suitable number of matched landmarks for which we have prior geometric data, we accurately estimate the position and orientation of the spacecraft with respect to the surface of the planetary body. We identify 3D to 2D point correspondences between our database and descent or orbital imagery. In the case of craters, the relevant structures are crater centroids, which is estimated carefully to account for perspective distortion effects. We study this accuracy of this approach via a detailed simulation. We set insertion altitude, orientation, imager resolution and FOV as desired. The onboard camera then takes a virtual snapshot of the visible portion of the 3D terrain. The 2D image coordinates of the crater centroids are distorted

noise with a truncated Gaussian distribution with predefined standard deviation and truncation point. If there are not enough craters (a tunable parameter n with a minimum value of 4) for position estimation, we mark the insertion point as a failure. Otherwise, we randomly select n of the visible craters for position estimation. Assuming matches between the 2D and 3D datasets, we use our position estimation algorithm to estimate spacecraft position and orientation and compare to the preset ground truth. For simulation purposes, we corrupt image data with 1 pixel noise. In Fig. 4 we show the result for position estimation for our core algorithm and for the statistically robust version. In each case, we assume 12 point matches and a 1 pixel random image noise with a variable percentage of outliers. The camera is assumed to be at 8 km above the surface.

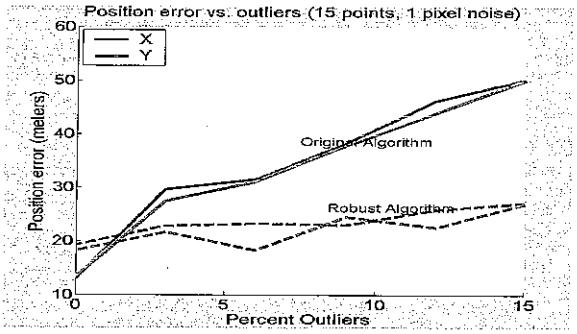

Figure 4: Position estimation error as a function of outliers in matched craters

Spacecraft Velocity Estimation Analysis

Prior to integration of a full Kalman filter solution for velocity estimation, we evaluate a simple estimation scheme based on integrated IMU acceleration data and position estimates provided by our algorithms above. Our analysis gives an upper bound for expected velocity error.

Without acceleration data, two camera frames provides an estimate of average velocity only. However, if acceleration is present, we can compute an instantaneous velocity, depending on IMU sampling an accuracy by simple integration. If the frames are taken at times to and t_f with recovered positions $P(t_O)$ and $P(t_f)$ we compute $V(t)$ from the acceleration a(t) as follows:

$$V(t) = \frac{P(t_f) - P(t_o) - \int_{t_o}^{t_f}\int_{t_o}^{t} a(\tau)d\tau}{t_f - t_o} + \int_{t_o}^{t} a(\tau)d\tau$$

If more than two frames are available, we take all pairs and compute a weighted sum. We have determined empirically that weighting linearly by time interval between frames and inversely by distance from the ground plane at time of frame capture works well. In Fig. 5., we plot the velocity error for a trial trajectory with varying frame numbers and frame intervals. With the current simulation framework in place, we will be able to do much more sophisticated analysis of velocity estimation using varying trajectories and error models for IMU and image noise and crater matching.

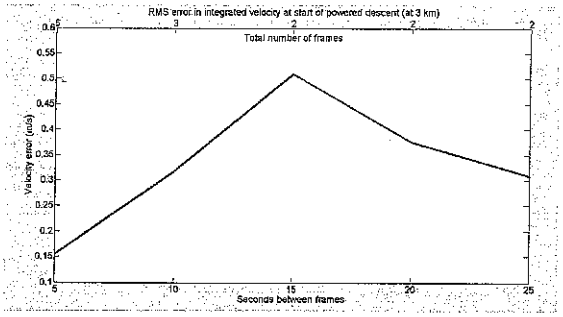

Figure. 5: Velocity estimation error using varying frame numbers and intervals.

With the simple scenario tested, we obtain velocity estimates at least an order of magnitude better than those currently available in mission scenarios.

Landing Error Analysis

We consider only the landing error arising from our position estimation system. We do not attempt to duplicate the sophisticated guidance algorithms to be used during a powered descent. Instead we report landing error arising from vision based position estimates as the sum of the position estimate at start of powered descent and the drift due to instantaneous velocity estimation error. Thus, if LE = landing error, PE = position estimation error at powered descent, VE = velocity error at powered descent, and t = time remaining to the ground, we have

$$LE = PE + VE * t$$

In Fig. 6 we show the landing error for the case of the MSL-like trajectory discussed above for various starting points for powered descent. Observe, that for this case of perfect acceleration, we are well within the requirements for pinpoint landing with integrated IMU data.

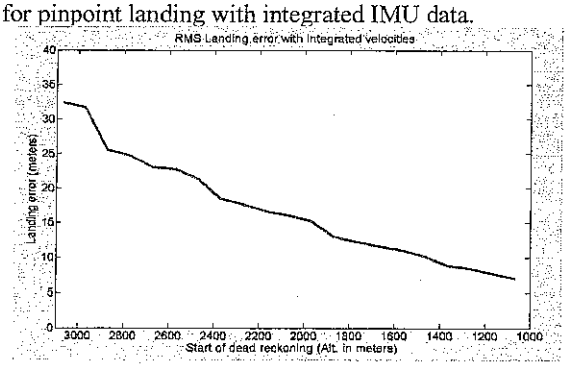

Figure 6: Landing error for MSL-like trajectory assuming IMU integrated (green)

If the base map errors (both position and elevation) are also considered, the landing error increases as expected.

Fig. 7 and 8 show the landing error vs map error. We see that the landing error is well within the requirements of pinpoint landing if the map and elevation error are less than 5 meters.

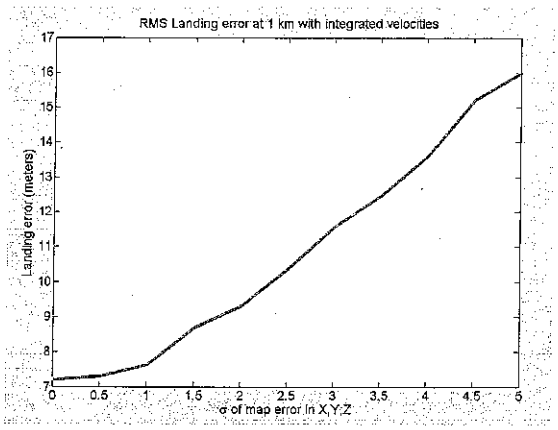

Figure 7: RMS landing error vs. map error when the spacecraft starts dead reckoning at 1 km.

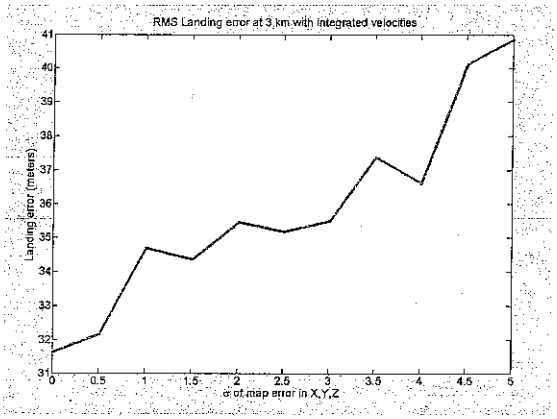

Figure 8: RMS landing error vs. map error when the spacecraft starts dead reckoning at 3 km.

IV. CONCLUSIONS

In this paper, a landmark (crater) based position estimation system is suggested. The performance analysis shows that this system is able to guild a spacecraft to land on Mars within 100 meter from the targeted landing site under normal noise conditions.

Acknowledgments

The research described in this paper was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

References

[1] Yang Cheng, A. Johnson, C. Olson, L. Matthes, "Optical Landmark Detection for Spacecraft Navigation" 13th Annual AAS/AIAA Space Flight Mechanics Meeting.

[2] Yang Cheng, James Miller "Autonomous landmark Based Spacecraft Navigation", 13th Annual AAS/AIAA Space Flight Mechanics Meeting.

[3] J. K. Miller et al "Determination of Shape, Gravity, and Rotational State of Asteroid 433 Eros", *Icarus* 155 3-17 (2002).

[4] David Forsyth et all, "Invariant descriptors for 3-D object recognition and pose" *IEEE PAMI,* Vol. 13, No. 10, 1991.

[5] Gouglas Bernard and Matthew Golombek, "Crater and rockhazard modeling for Mars Landing", AIAA Space 2001 conference, Albuquerque, NM.

[6] A. Ansar and K. Daniilidis, "Linear Pose Estimation from Points and Lines," *IEEE PAMI*, 25:578-589, 2003.

[7] B.K.P. Horn, et al., "Closed-form Solution of Absolute Orientation Using Orthonormal Matrices," Journal Opt. Soc. Am. A, A5: 1127-1135, 1988.

[8] C-P. Lu, G. Hager, and E. Mjolsness, "Fast and Globally Convergent Pose Estimation from Video Images," *IEEE PAMI* 22(6): 610-622 (2000)

[9] Yang Cheng, et. al "The Mars Exploration Rovers Descent Image Motion Estimation System", IEEE *Intelligent Systems*, May/June 2004, 13-21

[10] W.K. Hartmann, "Martian Cratering, 6, Crater count isochrones and evidence for recent volcanism from Mars Global Surveyor," *Meteoritics and Planetary Science,* 34, 167-177, 1999.