An Analysis of Spacecraft Localization from Descent Image Data for Pinpoint Landing on Mars and Other Cratered Bodies

Adnan Ansar and Yang Cheng

Abstract

A pinpoint landing capability will be a critical component for many planned NASA missions to Mars and beyond. Implicit in the requirement is the ability to accurately localize the spacecraft with respect to the terrain during descent. In this paper, we present evidence that a vision-based solution using craters as landmarks is both practical and will meet the requirements of next generation missions. Our emphasis in this paper is on the feasibility of such a system in terms of (a) localization accuracy and (b) applicability to Martian terrain. We show that accuracy of well under 100 meters can be expected under suitable conditions. We also present a sensitivity analysis that makes an explicit connection between input data and robustness of our pose estimate. In addition, we present an analysis of the susceptibility of our technique to inherently ambiguous configurations of craters. We show that probability of failure due to such ambiguity is becoming increasingly small.

Introduction

Current planetary landing technology limits spacecraft delivery accuracies to tens of kilometers at best. For instance, the recent Mars Exploration Lander (MER) missions had a landing error ellipse of greater than 30 km × 80 km. Future missions call for pinpoint delivery of a lander to within 100 meters of a target pre-selected from orbital imagery. This will require a number of technological advancements in propulsion, guidance, navigation, and control (GNC), map generation, and terrain relative localization. Our work addresses the last of these areas by allowing for the first time localization accuracy on the order of tens of meters from the start of image acquisition at altitudes as high as 8 to 10 km above the surface.

Descent image-based spacecraft localization can be divided into three parts: (a) identification of landmarks in imagery, (b) matching of identified 2D landmarks to a database containing their 3D terrain relative coordinates, and (c) recovery of terrain relative spacecraft position, attitude and velocity from the 2D to 3D correspondences. The last problem is referred to in the Computer Vision community as *pose estimation*. The algorithms to solve each part of the localization problem must be efficient enough to run on a flight processor with very limited computational power within the time constraints of the mission. For a landing on Mars, this may be on the order of two to three seconds from image acquisition to spacecraft localization using a 100 MIPS PowerPC 750 processor. For the case of crater landmarks, we have demonstrated in Cheng and Ansar (2005) solutions to the identification and matching problems and have shown that our spacecraft localization algorithms run within the time limits for a Mars mission on flight equivalent hardware. Figure 1 shows some example crater matching results while Figure 2 presents runtimes for various system components on a PowerPC 750. The advantages of this crater landmark based system are:

1. Less sensitivity to illumination conditions and viewpoint change due to relative invariance of crater rims, hence centroids, over these changes. This relaxes the requirements for base map generation since the map can be created from different orbital sensors and at different times of day and image scales.
2. Less dependence on onboard spacecraft state estimation since the crater-matching algorithm uses crater geometry (shape, size ratio, distance ratio, and perspective invariance properties) directly to match craters between the descent image and the base map.
3. Minimal onboard data storage requirement. Each landmark (elliptical crater) is represented by nine parameters (3D position, major and minor axes, 2D orientation, and surface normal) and can be stored in 36 bytes in computer memory. The landmarks in a heavily cratered landing site can be stored in less than 100 kilobytes, which is two to three orders of magnitude smaller than a raster map.

These results are presented in detail in Cheng and Ansar (2005).

This paper focuses primarily on feasibility of the technique. It is organized into two parts. The first addresses localization accuracy given 2D image to 3D map correspondence and presents experiments to show applicability to future EDL scenarios. In this section, we also present a parametric analysis of the sensitivity of our method to general crater configurations. This is an essential tool to identify near-degenerate cases which can give rise to suspect pose estimates. The second part of the paper addresses in detail the likelihood of inherent geometric ambiguity in detected craters which can defeat pose estimation.

Localization Algorithms and Expected Accuracy

The problem of determining the six degree of freedom pose (position and attitude) of a calibrated camera from known

NASA Jet Propulsion Laboratory, California Institute of Technology, 4800 Oak Grove Drive, Pasadena, CA 91109-8099 (ansar@jpl.nasa.gov, ycheng@jpl.nasa.gov).

Photogrammetric Engineering & Remote Sensing
Vol. 71, No. 10, October 2005, pp. 1197–1204.

0099-1112/05/7110–1197/$3.00/0

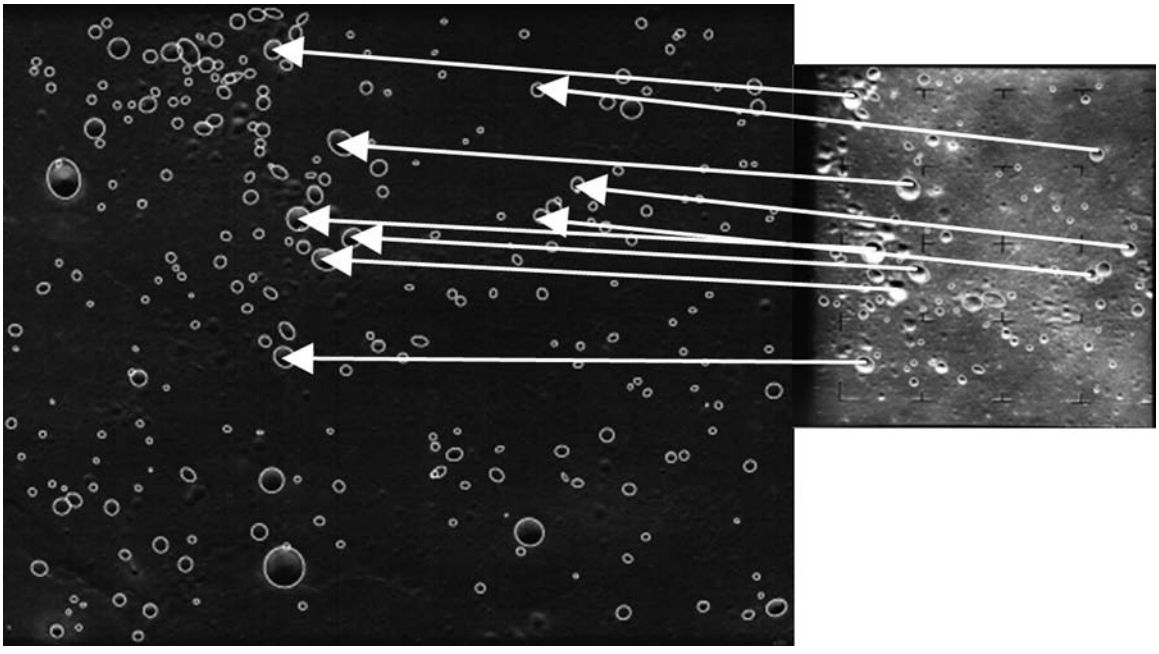
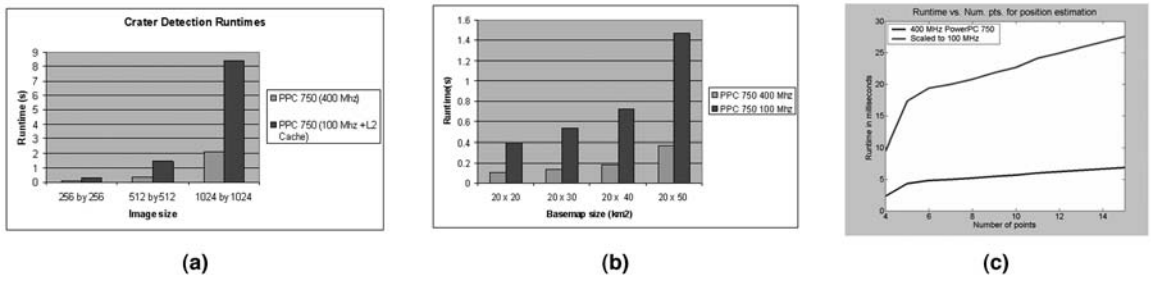


Figure 1. Example of the crater landmark based spacecraft localization system. (a) is Apollo 16 Panoramic Camera image, which serves as the base map, and (b) is a Ranger VII image. Images have different illumination and viewing angles. However, the system is able to match the craters between the two images.

correspondence between 3D points and their 2D image counterparts is classical in both photogrammetry and computer vision. In the former, it is known as space resection and in the latter as the *n*-point pose problem or pose estimation. Many solutions exist, most depending on non-linear optimization to minimize some cost function based on image re-projection error or 3D reconstruction error. Our approach consists of two steps, initialization and refinement. The first step uses a non-iterative, direct linear solution to the pose problem. It has a number of advantages. These include no need for initialization, no possibility of convergence to a local minimum, applicability to as few as four correspondences, and a guaranteed correct solution provided input noise is constrained. The main drawback of this algorithm is a highly non-linear complexity with respect to the size of the input data. Thus, we often use only a subset of the data to initialize. The refinement step consists of an iterative optimization algorithms combined with a statistically robust estimation scheme. The iterative algorithm scales well with the size of the input data, is relatively insensitive to noise, and is both highly accurate and fast converging given good initialization.

We begin with a few preliminary points of notation and nomenclature. Recovering camera pose corresponds to finding the Euclidean transformation $(\boldsymbol{R},\boldsymbol{T}) \varepsilon \mathbf{SO(3)} \times \boldsymbol{R}^3$ taking points in some global coordinate frame to those in a frame centered on the camera. If $\boldsymbol{V}_{\mathbf{w}}$ is a point in the global frame, then its representation in the camera frame is $\boldsymbol{V}_c = \boldsymbol{R}\boldsymbol{V}_w + \boldsymbol{T}$. The pose of the camera in the global frame is then given by $(\boldsymbol{R}^T, -\boldsymbol{R}^T\boldsymbol{T})$. The camera frame is always



(a) (b) (c)

Figure 2. The runtime of the three components on a PPC 750 computer. (a) the run times of the crater detection algorithm on different images sizes (b), The run time of the crater matching algorithm on different sizes of base map with map units in km, (c) The run time of the pose estimation algorithm on different number of points. The current system is able to do spacecraft localization in less than three seconds on a 512 × 512 image. This speed is adequate for the pinpoint landing application.

located at the center of projection with the z-axis coincident with the optical axis, and positive x to the right, and positive v downward on the image plane. Normalized coordinates refer to coordinates expressed in the camera frame with $z = 1$. These are obtained from pixel coordinates using the known intrinsic geometry of a calibrated camera.

Pose Initialization

We use a recent linear algorithm that provides localization accuracy nearly on a par with the best non-linear optimization schemes for cases of moderate noise in input data. As mentioned above, this approach avoids many of the usual problems associated with iterative algorithms, such as slow convergence and convergence to a local minimum. Details can be found in Ansar and Daniilidis (2003). We present only an overview here.

Suppose that n points in some global coordinate system are given by $\{\boldsymbol{w_i}\}$ and their normalized image coordinates by $\{\boldsymbol{p_i}\}$. Then there exist real numbers λ_i such that $\{\boldsymbol{q_i} = \lambda_i \boldsymbol{p_i}\}$ are the 3D coordinates of the points in the camera frame. Let d_{ij} be the squared distance between points $\boldsymbol{w_i}$ and $\boldsymbol{w_j}$. Since distance is a Euclidean invariant, we have

$$d_{ij} = (\lambda_i P_i - \lambda_j p_j)^T (\lambda_i P_i - \lambda_j p_j)$$

$$= \lambda_i^2 p_i^T p_j + \lambda_j^2 p_j^T p_j - 2\lambda_i \lambda_j p_i^T p_j. \quad (1)$$

This is a quadratic system of $n(n-1)/2$ equations in the variables $\{\lambda_i\}$. If we find an efficient solution for this system, we will have recovered the 3D coordinates of all points in the camera frame. Then, recovering the world to camera transformation is an instance of the classical absolute orientation problem. We solve the latter following Horn *et al.* (1988). The solution of the system in Equation 1 proceeds as follows. We replace the product $\lambda_i \lambda_j$ with the symbol λ_{ij}. Now Equation 1 is linear in the $n(n+1)/2$ variables $\{\lambda_{ij}\}$, and we are free to use linear algebra techniques such as singular value decomposition (SVD). Unfortunately, the linear system is under-constrained and has a multidimensional kernel. Suppose that the true solution is the vector $\lambda = (\lambda_{11}, \lambda_{12}, \ldots)^T$. If $\{\nu_i\}$ is a basis for the kernel of the original linear system, then there exist real numbers $\{k_i\}$ such that

$$\lambda = \Sigma k_i \nu_i. \quad (2)$$

The solution for the coefficients $\{k_i\}$ depends on the observation that for any quadruple of integers $\{a, b, c, d\}$ and any permutation $\{a', b', c', d'\}$, we have $\lambda_{ab}\lambda_{cd} = \lambda_{a'b'}\lambda_{c'd'}$. In effect, we are reintroducing the quadratic nature of the problem with these constraints. By substituting rows of λ in Equation 2 into expressions of this form, we obtain another quadratic system in the k's. As before, we linearize this system by replacing second order terms with symbols. The new system is provably over-constrained, and we always obtain a unique solution. Once we have λ, we can obtain $\{\lambda_i = \mathrm{sqrt}(\lambda_{ii})\}$, and the 3D coordinates of all points are known in the camera frame.

The runtime for this algorithm is comparable to non-linear optimization techniques for up to eight points. Assuming we have matched more than eight points during descent, we take a random sampling of less than eight to initialize with the algorithm just described.

Pose Refinement

We now give a brief overview of the refinement step. This combines a fast converging iterative algorithm from Lu, Hager, and Mjolsness (2000) with some standard robust statistical techniques. The algorithm of Lu, *et al.* (2000) is guaranteed to converge but may converge to a local minimum if poorly initialized. Hence, we start its search at the result provided by our initialization step above. The novelty of this algorithm lies in its use of an objective function in space rather than on the image plane. This is similar in spirit to the constraint in Equation 1. However, instead of an explicit 3D reconstruction, the technique relies on what Lu *et al.* (2000) refer to as object space colinearity. Essentially, they minimize the distance between the line of sight vector associated with a 2D point and its 3D counterpart. If $\boldsymbol{p_i}$ is again an image point in normalized coordinates, then consider the matrix

$$P_i = (p_i p_i^T)/(p_i^T p_i). \quad (3)$$

We assert without proof that this matrix projects any point in space onto the ray defined by $\boldsymbol{p_i}$. Then using the notation of this and the previous section, the objective function used to find $(\boldsymbol{R,T})$ is given by

$$\mathbf{E}(\boldsymbol{R,T}) = \Sigma \|(\boldsymbol{I} - \boldsymbol{P_i})(\boldsymbol{R}\boldsymbol{w_i} + \boldsymbol{T})\| \quad (4)$$

where $\boldsymbol{P_i}$ is given in Equation 3. Lu *et al.* (2000) present an elegant iterative scheme based on Horn's solution of the absolute orientation problem. This requires an SVD computation of a 3×3 matrix at each step, so that the complexity grows very slowly as a function of data size and is limited by construction of a 3×3 cross covariance matrix from n points.

We combine this algorithm with a Least Median of Squares (LMEDS) estimator to provide outlier rejection and a statistically robust solution. Given n matched landmarks, we determine pose from a subset of size $m < n$ a total of L times. L is chosen to guarantee a maximum algorithm failure rate of P_{fail} given m landmarks and a probability P_g that any given landmark is valid (i.e., not an outlier). L is given by

$$L = \log(P_{fail})/\log(1.0 - P_g^m). \quad (5)$$

Details and justification can be found in Fischler and Boller (1981). For each trial, we compute the median error in reprojection of 3D points onto viewing rays as given by Equation 3. We then take as a valid model the trial with lowest squared median error. Any data point exceeding an error threshold with respect to this model is considered an outlier. Finally, we take all points that are not outliers, and apply the core pose estimation algorithm to obtain the best pose.

Note that while the results presented in this paper assume cratered terrain, the localization algorithms will work equally well with other features, provided accurate 2D image to 3D map correspondences can be established. We are currently exploring such features for navigation around small bodies (asteroids and comets). Details of ongoing work were presented in Ansar (2004).

Pose Accuracy, Experiments, and Results

We have conducted numerous experiments using Mars-like as well as other scenarios to test the accuracy of our localization algorithms and have developed a full simulation environment for further testing. We now present two experiments to demonstrate the accuracy and robustness of our technique. The first is a purely synthetic result showing localization error in simulation. The second show a Mars landing scenario using real imagery from the Mars Odyssey THEMIS instrument.

Experiment 1

We have developed a Monte Carlo simulation to characterize the performance of our pose estimation technique subject to varying noise levels in both imagery and 3D map knowledge,

changes in camera geometry (field of view (FOV), sensor resolution, spacecraft insertion altitude and attitude, and probability of outliers in matched 2D and/or 3D data. The purpose of this simulation is to enable a parametric study of localization error. We decouple this from the image processing problem by avoiding the use of synthetic imagery and supplying directly the 3D coordinates of synthetic data points, projecting to the image plane and corrupting with noise or introducing outliers. In Figure 3, we show the result for lateral localization error of a spacecraft at an altitude of 8 km with 100 degree FOV imager at 1024 $\times$ 1024 resolution (equivalent to approximately 19 m/pixel on the ground) and an assumption of 15 matched data points. Systematic Gaussian image noise is added at a level of one pixel standard deviation, which we expect is higher than we will find in practice. In this example, we vary the percentage of outliers from 0 percent to 15 percent. A point marked as an outlier is selected with uniform probability from a 30 pixel $\times$ 30 pixel box centered at ground truth. The results displayed are mean values over 1000 trials. We show results for both the LMEDS robust algorithm and a version using only the initialization and iterative refinement step. Observe that the LMEDS output remains stable over outlier rates much larger than $<$5 percent rates we typically expect.

The results are fully consistent with the EDL requirements of $<$100 meters localization accuracy for upcoming missions. Note that while we present only lateral position error, which is of primary relevance to pinpoint landing, we in fact solve for the full six degrees of freedom pose of the spacecraft. While there is some inevitable coupling of position and attitude in the solution, this effect is generally minor, and the primary error component in the algorithm lies in the position estimate. See Ansar and Daniilidis (2003) for a complete discussion.

Experiment 2

We have developed a simulation environment that allows us to test our crater detection, matching and pose algorithms in an end-to-end fashion through a full landing trajectory. We demonstrate its application to a Mars landing scenario using a trajectory based on specifications for the planned Mars Science Lab (MSL) mission. The 3D database (map) consists of a large composite image of Martian terrain containing over 790 craters taken by the Mars Odyssey THEMIS instrument. Using the descent trajectory mentioned, we generated synthetic imagery to simulate the view from a lander. Note that this is a preliminary dataset used for development of the simulator, and the same image is used for the map, and then zoomed and warped according to ground truth spacecraft pose to produce the descent views. Thus, while this dataset does demonstrate invariance of our detection and matching algorithms to geometric distortion, it does not demonstrate photometric invariance. However, we have ample evidence from the NEAR mission that our crater detection and matching algorithms work well with lighting changes under real mission conditions. See Cheng and Miller (2003) for details. In future, we will use different base images for map and descent imagery generation in our simulator.

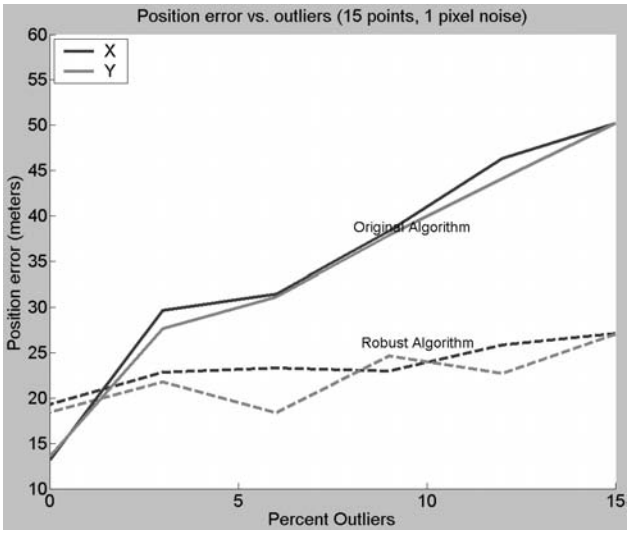


Figure 3. Lateral position error at 8 km. altitude for varying probabilities of outliers. Note that the robust algorithm remains stable and produces estimates are well within the requirements of future Mars missions.

Since the Martian surface as viewed from high altitudes is essentially planar, we used a homography transform to warp the map image and generate the descent views. Planarity guarantees that this is a faithful and realistic representation of the view from a descent camera with 512 $\times$ 512 imager and 90 degree FOV. In Figure 4, we show a snapshot from our simulator. The three panels on the upper half of the display show camera view, detected craters, and craters matched to the map, respectively. The lower left panel shows the true 3D position of the spacecraft in dark gray and the estimated position in white. The lower right shows absolute position error. Throughout this trajectory from 12 km to 8 km in altitude, the position error never exceeds 50 meters in norm, and the final error at 8 km altitude was 9 meters for the run displayed.

The 8 km altitude limit is a function of the 18 m/pixel resolution of the THEMIS image the 512 $\times$ 512 resolution of our descent camera. With a 1000 $\times$ 1000 camera with 90 degree FOV and map resolution on the order of Mars Reconnaissance Orbiter (MRO) of 30 cm/pixel, our algorithm should be useable to an altitude of approximately 150 m, provided there are suitable craters in the view. Assuming a more realistic lower limit of 50 m for the smallest visible craters, our technique with an MRO-like map will operate reliably to within 1 km of the surface.

Note that we are using our algorithms for crater detection and matching on a real image of Mars. The position estimation uses the full robust version of our algorithm and estimates all six degrees of freedom of the spacecraft pose; no prior information on altitude or attitude is assumed at any time. Observe that noise is not artificially introduced but is a real artifact of image processing.

Covariance Propagation

A useful pose estimation system must guarantee correct pose to within a required accuracy and with predictable statistics. This depends not only on the pose algorithm and the accuracy of landmark detection and matching in image data (e.g., our crater search algorithms) but on the distribution and accuracy of the positions of these landmarks in space. There are configurations that lead to especially sensitive solutions. We must be able to recognize these when they occur, and characterize the uncertainty in resulting pose estimates. At the same time, we must be able to characterize analytically the effects of uncertainty in 3D knowledge.

Propagation of covariance to estimated pose parameters is needed to provide an essential confidence metric. Without this information, risk mitigation is impossible. Furthermore, any sophisticated filtering technique for state estimation requires a covariance matrix for estimated parameters. Even

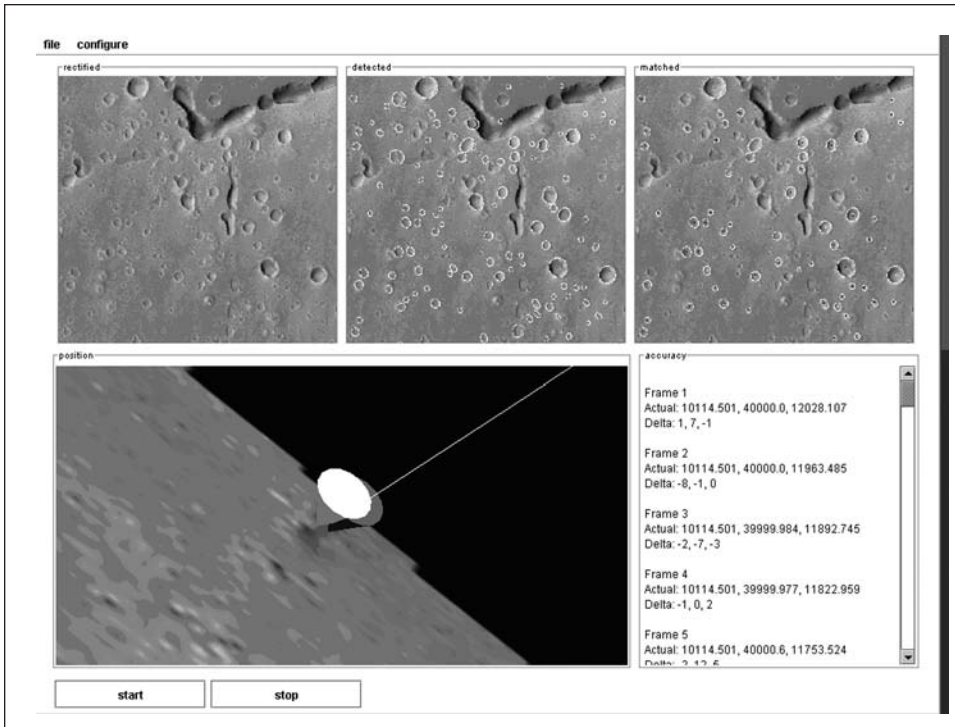


Figure 4. Snapshot from crater detection/position estimation simulator. The upper panes show the output of the crater detection and matching algorithms. The lower left pane shows the ground truth position (dark cone) and estimated position (light cone) of the spacecraft. The lower right pane shows position error in meters.

in cases where noise levels are low, there are likely to be configurations of landmarks that are inherently noise sensitive. It is critical that such situations be identified when they occur and that the directions of most- and least-stable pose be identified.

We follow the technique described in Haralick (1994) to propagate covariance from error estimates on 3D and 2D knowledge. The technique depends on existence of an objective function that is directly minimized to obtain estimated parameters from measurements. This constraint is used to solve implicitly for errors in the former given errors in the latter. Propagation is correct to first order. While our initialization algorithm does not directly minimize the constraint in Equation 1, the refinement algorithm does directly minimize the constraint in Equation 4. Thus, a covariance on overall pose can be determined.

Given a constraint to be minimized of the form $f(\theta, X) = 0$, where θ are parameters to be estimated and X is a set of measurements (in our case, 2D image measurements and 3D map coordinates), we calculate an implicit error $\Delta\theta$ on the estimates from an error ΔX on the measurements. In the presence of noise, f achieves a minimum at $f(\theta + \Delta\theta, X + \Delta X)$. Thus,

$$\frac{\partial f}{\partial \theta}(\theta + \Delta\theta, X + \Delta X) = 0. \tag{6}$$

Now, taking a Taylor expansion of $\partial f/\partial\theta$ about (θ, X) and observing that the gradient also vanishes at this point, we obtain to first order the relationship

$$\left(\frac{\partial f^2}{\partial X \partial \theta}\right)^T \Delta X + \left(\frac{\partial f^2}{\partial^2 \theta}\right)^T \Delta\theta = 0. \tag{7}$$

From this we can solve directly for $\Delta\theta$ in terms of ΔX. The covariance of θ is simply $E(\Delta\theta * \Delta\theta^T)$, where E signifies the expectation value. Similarly, we have $Cov(X) = E(\Delta X * \Delta X^T)$. A trivial substitution into Equation 7 then gives us

$$Cov(\theta) = \left(\frac{\partial f^2}{\partial^2 \theta}\right)^{-T} \left(\frac{\partial f^2}{\partial X \partial \theta}\right)^T Cov(X) \left(\frac{\partial f^2}{\partial X \partial \theta}\right) \left(\frac{\partial f^2}{\partial^2 \theta}\right)^{-1}. \tag{8}$$

For simplicity, we have used numerical derivatives in all computations of Jacobians. In Figure 5, we show the 1σ error ellipses for both position and attitude for several cases of synthetic data using the analysis above. We vary the image noise, 3D map noise, image point distributions and number of detected points. The resulting error ellipses behave according to intuition. Larger numbers of points shrink the ellipses. Higher noise levels cause them to grow. Biases in the distribution reflect in obvious ways.

Cov(X) is currently just a diagonal matrix encoding expected image and map noise variances. However, a full covariance matrix can be accommodated if additional structure in measurement error is known. Observe that the maximum error directions clearly change as a function of crater/landmark distribution. Note also that the magnitudes of the position errors are well within the requirements for precision landing for high noise levels. Attitude errors remain well under 0.5 degrees throughout.

Compensation for Velocity

In a real landing, it is not sufficient to find instantaneous position and attitude only. Course corrections must incorporate ground relative velocity information as well, since the effect of winds cannot be determined from IMU data. An image based horizontal velocity estimator, the Descent Image

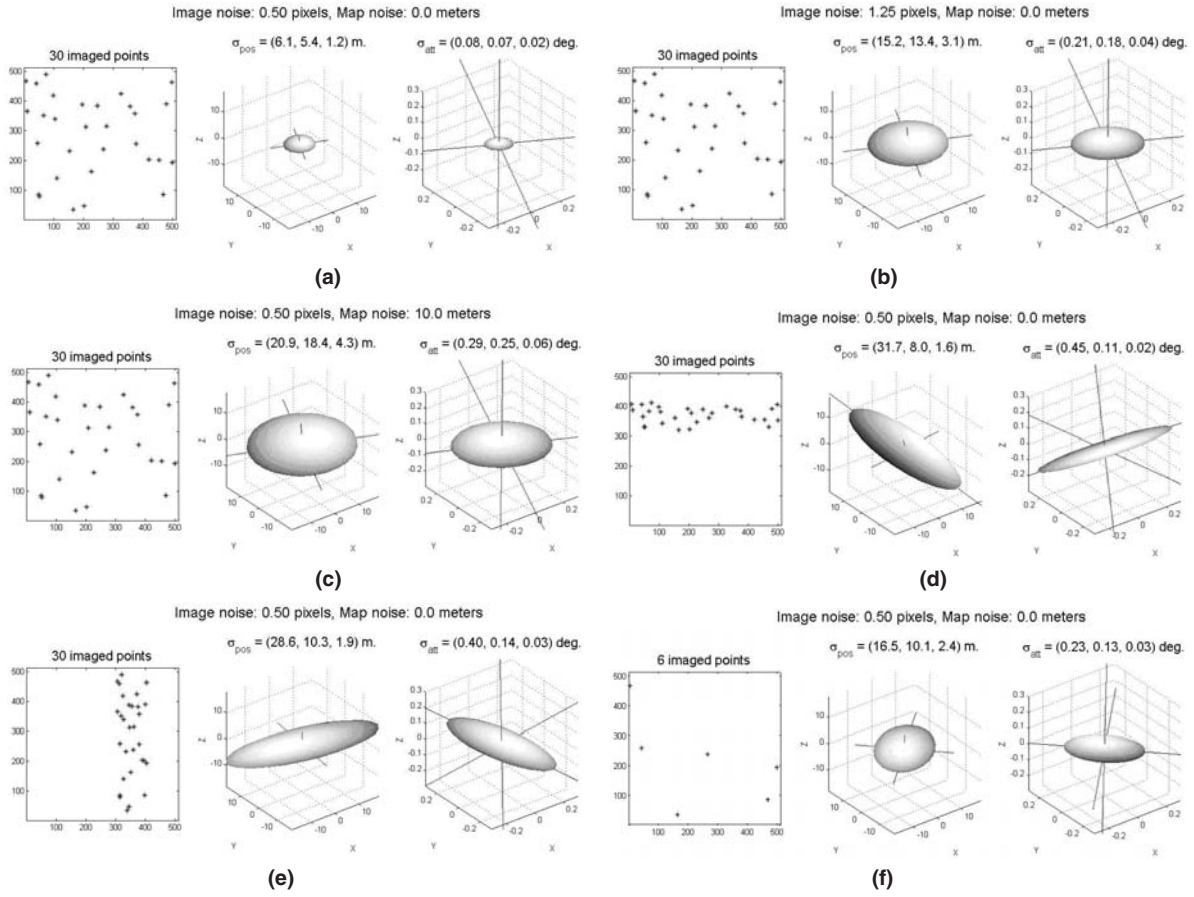


Figure 5. Each frame shows the synthetic image, position error ellipse and attitude error ellipse. All positions are bounded by 10 m in all directions and all attitudes by 0.2 degrees in all directions. (a) Moderate image noise (0.5 pixel) and perfect 3D map. (b) High image noise (1.25 pixel) and perfect map. (c) Moderate image noise and high map noise (10 meters). (d) Moderate noise and vertically biased distribution. (e) Moderate noise and horizontally biased distribution. (f) Low number of detected points (six) versus high number in other cases (30).

Motion Estimation System (DIMES), was successfully used on both Mars Exploration Rovers (MER) landings. The DIMES algorithm combines measurements from a descent camera, a radar altimeter and an inertial measurement unit (IMU). To deal with large changes in scale and orientation between descent images, the algorithm uses altitude and attitude measurements to rectify image data to a level ground plane. Feature selection and tracking is employed in the rectified data to compute the horizontal motion between images Cheng, *et al.* (2004). Although DIMES satisfied the MER landing requirement, it estimates only an average velocity from image data. This is inadequate for pinpoint landing applications. However, we show that given our position estimates, we can integrate IMU data to obtain "instantaneous" velocity estimates and bound landing error to well withing future mission requirements.

We estimate an upper bound on the landing error e_l as follows:

$$e_l = e_p + e_v * dT \tag{9}$$

where e_P is the position error at the moment of transition from freefall to powered descent, e_V is the instantaneous velocity error at that moment, and dT is the time remaining to the ground. Our effort here is a proof of concept only. We fully expect that any final system will incorporate a sophisticated Kalman or similar filter to determine velocity from our position measurements and easily surpass the results of our approach. However, even the relatively simple technique we describe below produces adequate results. Suppose two frames are taken at times t_f and t_o, with associated position estimates $P(t_f)$ and $P(t_o)$. If acceleration data $a(t)$ are also available, then a simple derivation shows that at any time t, the instantaneous velocity can be computed as

$$V(t) = \frac{P(t_f) - P(t_0) - \int_{t_o}^{t_f}\int_{t_o}^{t} a(\tau)d\tau}{t_f - t_o} + \int_{t_o}^{t} a(\tau)d\tau. \tag{10}$$

If multiple frames are available, we can further refine the estimate of $V(t)$ by taking a weighted average of all pairwise estimates. We have determined empirically that weighting by difference in time between the frames and inversely by altitude works well. So that if a sequence of indexed frames is available, then the sum over all $i < j$ of the quantity

$$V(t) = \sum V_{i,j}(t)\,\frac{t_f - t_i}{Z(t_f)} \Big/ \sum \frac{t_f - t_i}{Z(t_f)}. \tag{11}$$

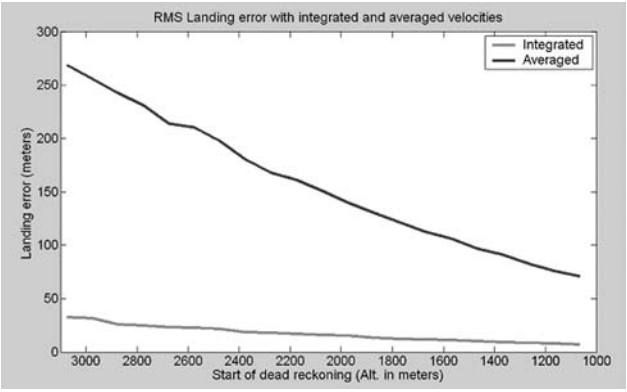


Figure 6. Comparison between averaged and integrated velocities for varying altitudes at which transition from freefall to powered descent starts. The integrated approach meets requirements for pinpoint landing on future Mars missions.

Here $V_{i,j}(t)$ is the instantaneous estimate of Equation 10 given by frames i and j. In Figure 6, we show the difference between using this integration technique in estimating e_l in Equation 9 versus an approach that estimates average velocity from the last two frames before transition to powered descent. Results are shown as a function of the altitude at which dead reckoning (i.e., transition from freefall to powered descent) starts. Note that while the averaging technique fails to meet requirements for pinpoint landing, the integrated technique easily satisfies them for all transition points considered.

Attitude Recovery

We have made little mention of attitude because position recovery is the more relevant issue for localization. In real mission scenarios, attitude is generally well known via star trackers prior to entry. Nevertheless, we always solve directly for all six degrees of freedom of the spacecraft pose. In the next section, we will show that attitude is very well constrained by our technique. In all cases discussed so far, we are able to estimate it to within 0.5-degree absolute error, even in the presence of high noise.

Crater Uniqueness Analysis

We now address the feasibility of our technique for the specific case of crater features. We have developed models for the likelihood of ambiguous crater configurations given specific crater size distribution models. These follow from purely geometric arguments under the assumption that all craters are circular. Naturally, this is a worst-case scenario, since eccentricity is a disambiguating factor. We validate our geometric arguments using real data from an Odyssey THEMIS image containing nearly 1,000 craters. Note that this study focuses exclusively on ambiguity within a pre-computed ground map. The spacecraft and imager used during descent are not factors. We now sketch the analysis.

A formal counting argument and some calculus shows that the probability of having two pairs of craters with centers at the same relative distance d in a disk of radius R is

$$P_1 = \left(\frac{3}{4}\frac{1}{R}\arctan\left(\sqrt{\frac{4R^2}{d^2} - 1}\right)\right) - \left(\left(\frac{d}{4}\right)^2 \frac{3}{R^3}\sqrt{\frac{4R^2}{d^2} - 1}\right). \quad (12)$$

Suppose two distances are considered "equal" provided they are within δ_p of one another. We can integrate Equation 12 subject to this definition of equality to find the probability that any two pairs of craters are at the same relative distance. Let this quantity be called P_2. Once a crater pair is chosen, we have implicitly fixed an origin and orientation on the full collection of craters. Suppose there are N craters in our region of interest and that m of them are used for pose determination. This leaves $n = N - m$ remaining. Now a further counting argument applies to determine the probability P_{geom} of having a second collection of m craters with a configuration geometrically identical to the first:

$$P_{geom} = (1 - (1 - \frac{\delta_p^2}{R^2})^n)^{m-2}(1 - (1 - P_2)^{(2n-3)(2m-3)}). \quad (13)$$

By geometrically identical, we mean that the two collections can be mapped onto one another by a planar Euclidean motion with less than δ_p error between any two corresponding crater centers.

So far, we have ignored crater size. We find in Hartmann (1999) a model for crater distribution on Mars. An equivalent representation for the size-frequency distribution function can be written as:

$$n = Kd^{-\alpha} \quad (14)$$

which relates the number n of craters per square kilometer to their size d subject to two parameters α and K. If we integrate this expression over all sizes from d_{min} to d_{max}, we find an expression for the total number N of craters in a region with area A. Then, the probability distribution for craters within the size range d_{min} to d_{max} is then given by $\sigma_N(d_{min},d_{max}) = N/A$. We make this explicit:

$$\sigma_N(d_{\min},d_{\max}) = N/A = \frac{K}{1-\alpha}(d_{\max}^{1-\alpha} - d_{\min}^{1-\alpha}). \quad (15)$$

Now if we set δ_s as the tolerance for "identical" crater size, the probability of two craters having identical size is found by integrating $\sigma_N(d, d + \delta_s)^2$ over all possible d. We call this quantity P_3. Then with yet another simple counting argument, we find the probability of having m craters of the same size out of the full collection to be

$$P_{size} = (1 - (1 - P_3)^n)^m. \quad (16)$$

Finally, the probability of an ambiguous configuration of m craters out of N is given by

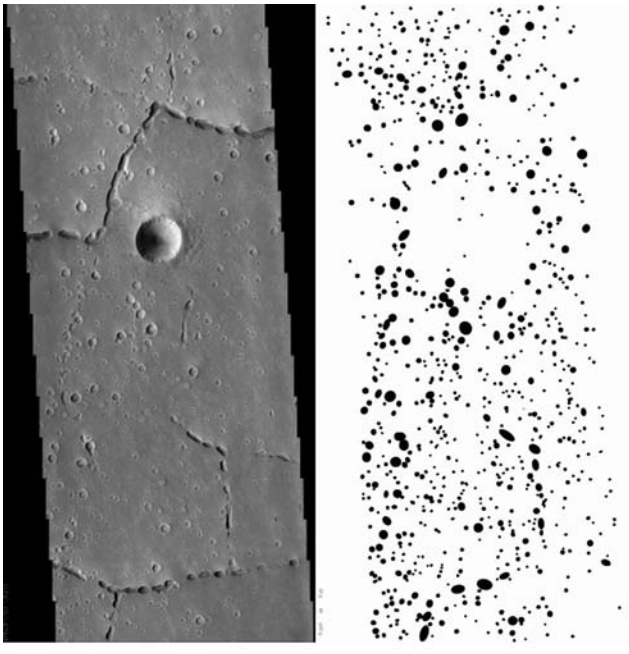
$$P_{ambiguous} = P_{geom}P_{size}. \quad (17)$$

Using information in Hartmann (1999), we derived plausible (K, α) values for lightly ($K = 0.1$, $\alpha = 1.8$), and heavily ($K = 0.3$, $\alpha = 1.8$) cratered terrain. Assuming 90 degree FOV and 8 km camera altitude, we set the diameter of our region of interest to 8 km to cover roughly four times the instantaneous camera view, $\delta_s = \delta_p = 30$ meters, $d_{max} = 4$ km, and $d_{min} = 100$ meters. Since our goal is to study ambiguity on the ground only, we assume that spacecraft location is known sufficiently well to restrict the valid map to this region. In Table 1, we show the results for $m = 2$ to 5. The probability of ambiguous configuration is already increasingly small for $m = 5$.

We verify these results with a real Mars image from Odyssey THEMIS and an exhaustive search over 917 identified craters for ambiguous configurations. An image of the terrain and the detected craters is shown in Figure 7. Note that this is the same image as used in Experiment 2. The difference in total number (790 in Experiment 2 versus 917 here) of craters is due to selection of a larger portion of the image in this test as well as a slight change in crater detection parameters. Detected craters range in size from 55 m in diameter to 457 m, and the

TABLE 1. PERCENT PROBABILITY OF AMBIGUOUS CONFIGURATIONS FROM GEOMETRIC ARGUMENT

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}



(a) (b)

Figure 7. (a) Mars terrain image taken from Odyssey THEMIS and (b) detected craters used to build 3D map.

TABLE 2. PROBABILITY OF AMBIGUOUS CONFIGURATIONS USING ODYSSEY THEMIS DATA

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}

scene covers a 45 km $\times$ 22 km patch of the Martian surface. Observe that the large crater near the center of the image is not detected because its size exceeded our search threshold, not because of some inherent deficiency in the algorithm.

The results for $m = 2$ to 5 are given in Table 2. Observe that they fall squarely between the lightly and heavily cratered cases obtained by the geometric argument above up to order of magnitude. We believe that this validates our counting argument and gives further credence to the low probability of ambiguous configuration.

Conclusions

We have presented an image based localization technology that satisfies the requirements for future NASA missions requiring pinpoint landing with emphasis on feasibility in terms of localization accuracy and applicability to Martian terrain. Along with the detailed description of detection and matching algorithms reported elsewhere, this provides the first plausible solution for autonomous spacecraft localization during descent. Our solution is fast, robust and requires few identified landmarks to produce an accurate estimate. We have shown that the technique does not suffer from inherent ambiguity under even mild assumptions. Furthermore, we have developed a covariance model parameterized by image and map noise. This provides a tool for studying position estimation as a function of crater/landmark distribution as well as a real-time confidence measure on the estimate obtained from a given dataset.

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. We thank Siqi Chen for help in developing the simulator pictured in Figure 4.

References

Ansar, A., 2004 Small Body GN&C Research Report: Feature Recognition Algorithms, JPL Internal Doument D-30282 in Small Body Guidance Navigation and Control FY 2004 RTD Annual Report, *JPL Internal Document D-30714*, pp. 151–171.

Ansar, A., and K. Daniilidis, 2003. Linear Pose Estimation from Points and Lines, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25:578–589.

Cheng, Y., and A. Ansar, 2005. Landmark Based Position Estimation for Pinpoint Landing on Mars, *Proceedings of the IEEE Conference on Robotics and Automation*, Barcelona, Spain.

Cheng, Y., J. Goguen, A. Johnson, C. Leger, L. Matthies, A. San Martin, and R. Wilson, 2004, The Mars Exploration Rovers Descent Image Motion Estimation System, *IEEE Intellegence Systems*, May/June: 13–21.

Cheng, Y., and J. Miller, 2003. Autonomous Landmark Based Spacecraft Navigation, *13th Annual AAS/AIAA Space Flight Mechanics Meeting*.

Fischler, M., and R. Bolles, 1981. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography, *Communications of the ACM*, 24:381–395.

Haralick, R.M., 1994. Propagating Covariance in Computer Vision, *Proceedings of the 12th International Conference on Pattern Recognition*, pp. 493–498.

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

Horn, B.K.P., H.M. Hilden, and S. Negahdaripur, 1988. Closed-form solution of absolute orientation using orthonormal matrices, *Journal of the Optical Society of America*, 5:1127–1135.

Lu, C-P., G. Hager, and E. Mjolsness, 2000. Fast and globally convergent pose estimation from video images, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:610–622.