

AUTONOMOUS LANDMARK BASED SPACECRAFT NAVIGATION SYSTEM

Yang Cheng*
and
James K. Miller**

An autonomous landmark based spacecraft navigation scheme is presented. This new scheme involves the following data processing steps: image selection and planning; landmark (crater) detection; preliminary image matching; crater matching; data base management; and finally landmark based orbit determination. This is the first truly autonomous landmark based navigation system. Compared with the conventional method, which was used for NEAR mission operations, this system is faster (>100 times faster), more reliable and more accurate. The system has been successfully tested on the 200 km NEAR orbit imagery. In this test, 100 m accuracy was obtained after an hour of processing, which required several weeks of processing using manual crater detection and matching.

Introduction

Optical landmark navigation using craters on the surface of a central body was first used operationally by the Near Earth Asteroid Rendezvous (NEAR) mission. This mission required the determination of Eros shape, gravity, rotation state and ephemeris as well as the spacecraft orbit and the primary data was optical imaging of craters on the surface of Eros. After an initial determination of Eros's physical parameters and rotation state which took about one month, the mission settled into a year of spacecraft navigation using landmark tracking in conjunction with laser altimetry and radiometric data. At the end of the mission, the spacecraft soft-landed on Eros. NEAR mission's success has proven landmark (crater) tracking is a powerful data type for spacecraft orbit determination in low altitude orbits. Tracking individual craters enables orbit determination accuracies on the order of the camera resolution, several meters for small bodies or several kilometers for terrestrial planets. This exceeds the accuracy that can be obtained from radiometric data alone.

* Member of the Mobility Systems Concept Development Section, Jet Propulsion Laboratory

** Member of Navigation Systems Section, Jet Propulsion Laboratory, Associate Fellow AIAA

The landmark tracking algorithm is contained in computer programs that are executed in sequence to produce a data point file suitable for input to existing orbit determination programs. The input data consists of images of Eros from several weeks in the past to several days in the future. A spacecraft ephemeris along with Eros's attitude, ephemeris and physical data including gravity field is available that is fit with high precision to the past data and is mapped several days into the future with moderate precision. At the conclusion of the data processing, a high precision trajectory is obtained that extends several days beyond the epoch of the last solution. The definition of high precision and moderate precision is that precision required for maintaining orbit stability and satisfying mission constraints and is a function of data accuracy and the number of days the orbit is mapped to the future.

Currently, the extraction of landmark data from images of a celestial body is performed by an algorithm that requires a large amount of human intervention, which is time consuming, tedious and sometimes unmanageable. For example, during the NEAR mission, two image analysts were dedicated to landmark processing for about two-years of mission operation. One of the most time consuming tasks was to extract and identify craters manually from thousands of NEAR optical images. About 10,000 craters were identified on Eros's 32-kilometer long and 16-kilometer wide body. It can easily be imagined that if the central target is a much larger body, (e.g. Vesta or Ceres which are 960 kilometers and 520 kilometers in diameter and they are the primary targets of the 2006 DAWN mission.) the time required for optical data processing as well as the number of craters needed to be extracted will increase proportionally. The manual approach used for the NEAR mission, therefore, will be indeed problematical.

A fully autonomous navigation system would begin on approach to Eros and perform all the navigation functions needed to orbit and land on Eros. This would require artificial intelligence to anticipate all the possibilities and this capability is indeed far in the future. A more practical autonomous navigation system would take over the mundane navigation functions once Eros has been characterized. Therefore, the essential requirement on the landmark tracking algorithm described here is to acquire new data and update the spacecraft orbit from a previously well determined orbit and physical characterization of the asteroid Eros using the methods developed on NEAR.

In the past few years, some significant breakthroughs toward automating the landmark based optical navigation have been made in the areas of computer vision and spacecraft navigation. For instance, a very robust and powerful crater detection algorithm has been developed by the JPL Machine Vision Group [1]. In the mean time, some landmark based navigation filters, which were developed for the NEAR mission, were readily available for integration. With these breakthroughs, it is time to merge these algorithms into an autonomous landmark based navigation system. In this paper, we will present such a system. This new system involves the following data processing algorithms 1. Image selection and planning; 2. Crater detection algorithm; 3. Preliminary image

matching; 4. Crater matching algorithm; 5. Database management; 6. Orbit determination filter (Fig.1).

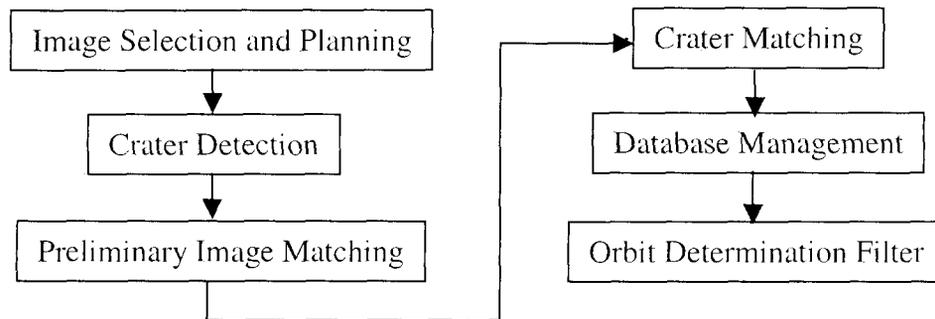


Figure 1: The autonomous landmark based navigation

IMAGE SELECTION AND PLANNING

The first computer program plans a set of images for landmark tracking. The spacecraft orbit is mapped several days into the future and a large number of images (several hundred) are taken of regions of Eros. The nadir point is imaged, if it is illuminated, as well as regions near the lit limb and terminator. These images are combined with any other images that may have been taken for science to form an image database. The sun direction and camera pointing are computed at the shutter time of each image from the *a priori* spacecraft orbit and Eros geometry. The *a priori* spacecraft orbit is obtained by mapping the previous orbit solution forward in time. The actual camera pointing, which is the spacecraft attitude since the camera is hard mounted to the spacecraft body, is obtained from telemetry. For each image, the shutter time, the planned camera pointing direction, the actual telemetered pointing direction and the sun direction relative to nadir are written to a file.

CRATER DETECTION

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. A typical crater in an image has an elliptical rim and a bright to dark shading pattern, which is dictated by the lighting azimuth and elevation as well as its own topography. These distinguishing characteristics are used extensively in the crater detection. The crater detection algorithm consists of five individual algorithms.

Edge Detection

This step detects edges in an image and places them in a database. The Canny Edge detection algorithm is used [3]. In order to extract both sharp and fuzzy rims, two rounds of edge detection are executed. In the first execution, a smaller kernel (< 5) is used, which aims to detect small and sharp edges. In the second round, a large kernel (~ 9) is

used to detect large and less sharp edges. Experimental study shows this strategy is very effective when image quality is poor such as the NEAR imagery. The image gradient (g_x, g_y) of each edge pixel is extracted. Both data sets are stored separately for future usage.

Rim Edge Grouping

This step groups together edges that belong to the same crater. The information used for this process include edge shape (convex), the image intensity profile inside a crater, edge gradients. If a pair of edges (lit and shaded side of crater) are found, they will be used to fit an ellipse.

Ellipse Fitting

This step fits an ellipse to each group of crater edges. A generic conic representation of an ellipse is

$$\begin{aligned} F(A, X) &= a^T X = \\ a x^2 + bxy + cy^2 + dx + ey + f &= 0 \end{aligned} \quad (1)$$

with constraint $4ac - b^2 = 1$

We use an iterative reweighting technique [4] for ellipse fitting.

Ellipse Refinement

This step adjusts the detected crater's geometry directly in the image domain to reduce errors introduced in edge detection and ellipse fitting. Consider an ellipse represented by five parameters: its center x_0, y_0 , major axis a , minor axis b and orientation t_0 . The equation of the ellipse under this parameterization is

$$\begin{aligned} x(t) &= a \cos t \cos t_0 - b \sin t \sin t_0 + x_0 \\ y(t) &= a \cos t \sin t_0 + b \sin t \cos t_0 + y_0 \end{aligned} \quad (2)$$

If the lighting direction is given, the image of the crater should satisfy a merit function

$$M = \frac{\oint \| (g \bullet n)(s \bullet n) \| dt}{\oint (x_t^2 + y_t^2) dt} = \max \quad (3)$$

where g is the image gradient at (x, y) , s is the lighting direction vector and n is the customized unit normal of crater rim as.

$$n(n_x, n_y) = \begin{cases} (x_u k_1 - x_t k_2, & y_u k_1 - y_t k_2) & \text{if } (n \bullet s) > 0 \\ (-x_u k_1 + x_t k_2, & -y_u k_1 + y_t k_2) & \text{otherwise} \end{cases}$$

We use a multidimensional iterative nonlinear minimization algorithm based on conjugate gradient to minimize this merit function to lock an ellipse precisely on the rim of a crater.

Crater Confidence Evaluation

The confidence metric is derived from the merit function given by Eq. 3. N points are evenly selected on both sides of a crater. No point is selected at the lit and shaded

transition area (30 degrees on each side) because of the large ambiguity in image gradients around those areas. The confidence of a crater is computed using

$$C = \frac{\sum_{i=1}^n K_i}{N}$$

$$K_i = \begin{cases} 1 & \text{if } (n_i \cdot g_i) > T \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

Where n_i is the customized normal of ellipse and g_i is the gradient at point i .

The intermediate and final result of this algorithm are shown in Figure 2. Fig. 2a is the original image, Fig.2b is the edge image by Canny edge detection; Fig 2c is the edge image after edge filtering and convex analysis. The black edges are the edges on the shaded side and the gray edges are the edges on the lit sides and Fig. 2d shows the detected craters. The more detailed description of this algorithm can be found in [1].

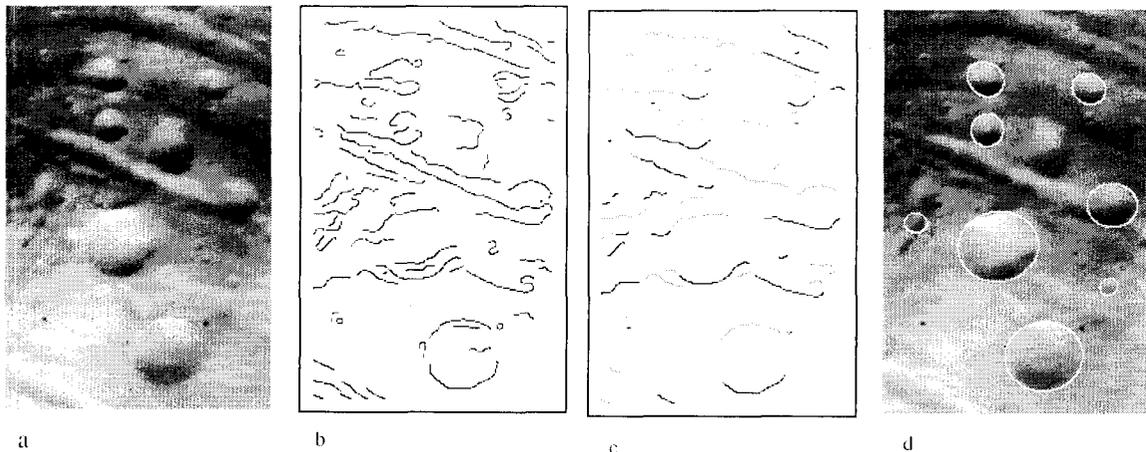


Figure 2 Some intermediate and final results of the crater detection algorithm.

This is an ideal algorithm for navigation orbit determination. It takes less than 5 seconds to detect ~12 craters from a 512 by 512 image on a 333 MHZ SUN Ultra 10 workstation, which is at least 100 times faster than manual crater delineation. Its detection rate is better than 90%, its false alarm rate is less than 5% and most importantly, its geometric error (position and shape) is less than a pixel.

The craters are numbered and written to a file along with the line and pixel coordinates of the crater centers and a set of geometric parameters. The geometric parameters consist of crater size, shape, orientation and lighting. The sun direction relative to nadir is used to aid crater detection and is obtained from the image selection and planning file.

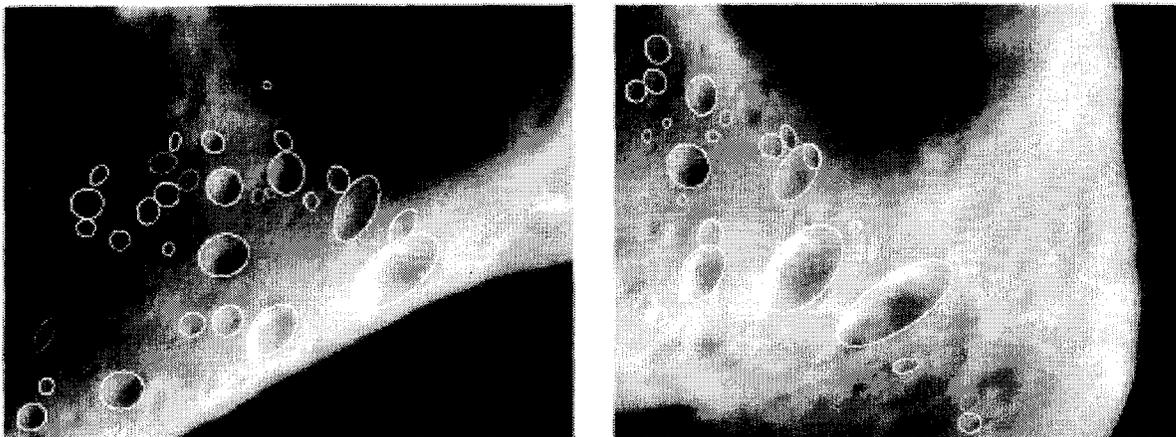


Figure 3 Crater detection results of NEAR images.

PRELIMINARY IMAGE MATCHING

The images that have landmarks identified by crater detection are put in time order and a file is written with the unique landmark number, image number, camera pointing and crater detection geometric parameters. An *a priori* shape model is used to compute *a priori* landmark locations that are also written to the same file along with the spacecraft and sun vector locations in Eros body fixed coordinates. The entire database is scanned and landmark pairs are identified as possible the same landmark on separate. The basis for this pairing is the computed landmark location and the uncertainty of computing the landmark location. The image pair numbers corresponding to the preliminary matched landmarks are written to a file. The purpose of this preliminary matching is to identify images whose lit regions overlap and are likely to have common craters.

CRATER MATCHING

The image pairs are scanned to determine all the landmarks on a pair of images that are common. For each image pair, a record is created in a file containing the two unique image numbers and the unique landmark numbers and line and pixel locations in each image. Parameters are set in the crater matching algorithm to ensure a high probability of obtaining a match. The size and shape, as well as the context or the position of the landmarks relative to other landmarks, are used to verify the match. This is difficult because a crater may have a very different appearance when it is viewed from different directions and with different sun angles. This is particularly true for the small and irregularly shaped bodies such as Eros. For example, Fig. 3 shows two images of the same area of Eros, where it is not easy to identify the same craters on the two images, even by visual inspection.

Three techniques are used here: image cross-correlation; context based matching [7]; and projective conic invariants [4,6]. These techniques are applicable when one observes an object from a distance that is an order of magnitude or more greater than the maximum

object diameter along the direction of view [4]. The optical navigation camera used on the NEAR mission has a 2.95 degrees horizontal by 1.34 degrees vertical field of view (FOV) and its minimum depth-to-width ratio is about 38, which is well below the threshold required by the algorithms above. However, none of these algorithms is perfect and all of them have advantages as well as disadvantages. The best possible solution is obtained by combining these algorithms into a single approach.

Cross Correlation Matching

When two images have similar photometrical properties and cover the same area, a crater matching can be confirmed by image cross correlation.

1. If a crater found in both images is assumed to be the same crater, the local surface normal is computed from the camera attitude and surface model;
2. Based on the spacecraft poses and local surface normal, the image relationship between the two images centered at the crater is approximated by

$$\begin{aligned} x_1 &= \frac{a_1 x_2 + a_2 y_2 + a_3}{a_7 x_2 + a_8 y_2 + 1} \\ y_1 &= \frac{a_4 x_2 + a_5 y_2 + a_6}{a_7 x_2 + a_8 y_2 + 1} \end{aligned} \quad (5)$$

3. A template centered at the crater is rectified using the above equation and it is placed in a window centered at the crater in the second image and at each pixel location a Pseudo-normalized correlation is computed:

$$cor = \frac{2 \sum (I_1 I_2)}{\sum I_1^2 + \sum I_2^2} \quad (6)$$

Where the I_1 and I_2 are the image intensity value of image 1 and 2.

4. Maximum correlation metric corresponds to the best estimate of template location in the window.
5. If the correlation (cor) is greater than a threshold (e.g. 0.9) and the location obtained from step 4 is very close to the center of the crater in image 2, their match is confirmed.
6. Otherwise, their match is rejected.

The cross correlation matching method works well even when only a few craters are found in the scene if the photometrical conditions are right and the approximate motion between them and the center target's geometry are known. However, when these conditions cannot be met, this method might not work. Another disadvantage of this approach is that it is slow.

Context Based Matching

Considering the positions and sizes of a set of craters found in an image, their constellation usually forms a unique pattern, which could be used to identify their correspondences in the second images.

Although the initial *a priori* spacecraft position and central body attitude may not be very accurate, they are useful for initiating crater matching. For example poses of two images can form an approximated epipolar geometry (Fig. 4). Assume C is a crater on an asteroid and two images at p_1 and p_2 cover C . C , p_1 and p_2 form an epipolar plane. The intersection between the epipolar plane and images plane are so called epipolar lines (solid lines in image planes). The images of crater C should lie on these lines if *a priori* geometry is accurate. If *a priori* geometry has a small error, the crater images should be near the epipolar line. The offset of a crater from the epipolar line can be used as an indicator of crater matches. Experimental studies show that the offset in the NEAR images is less than 20 pixels in general.

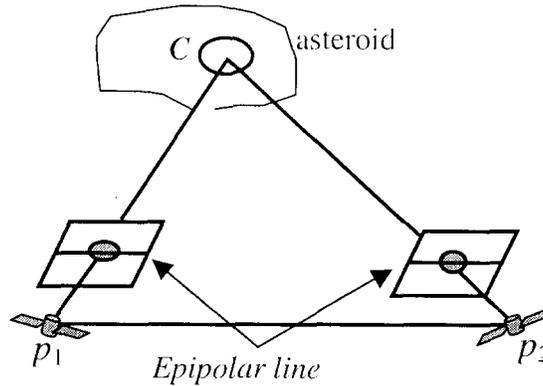


Figure 4. The epipolar geometry can be used for crater matching.

Another advantage of the epipolar geometry is that the epipolar geometry can be formed without knowing the 3D position of C . In this case, the vectors Cp_1 or Cp_2 can be directly computed from the crater center in images. Therefore, line p_1p_2 and Cp_1 or Cp_2 can determine the epipolar plane Cp_1p_2 .

Using homogeneous coordinates a conic C (Eq.1) can be represented as a quadratic form:

$$X^T C X = 0 \quad (7)$$

where

$$C = \begin{pmatrix} a & b/2 & d/2 \\ b/2 & c & e/2 \\ d/2 & e/2 & f \end{pmatrix} \quad (8)$$

Consider a linear transform $X' = TX$, the quadratic form after the transformation is

$$C' = T^T C T \quad (9)$$

The transform T can be obtained from both the spacecraft pose and the surface normal or from an initial guess of three conic pairs. A right transform T will be able to identify the corresponding conics (craters) between two image by comparing their positions and shapes (the quadratic form) (Fig. 5).

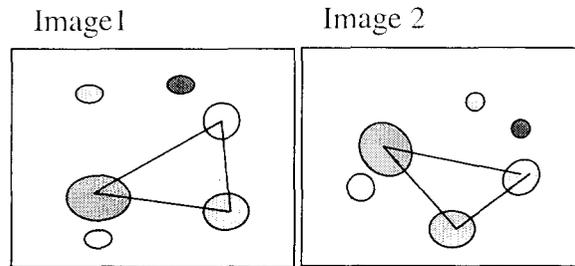


Figure 5: The crater constellation forms a unique pattern

1. Three craters are randomly selected in image 1. These three craters have to form a triangle in order to avoid an ill conditioned matrix computation (Fig. 5);
2. Based on the initial spacecraft attitude and position and one crater position in the first image, the epipolar geometry between the two images is generated. All craters, that are very close to the epipolar line and have a close size ratio in image 2, are selected.

$$e < E_i \quad S_1 < \frac{s'}{s} < S_2$$

where e the distance to the epipolar line and s' and s are the size of crater in image 1 and 2 respectively.

3. For each crater in image 1, each crater selected in step 2 could be a potential match. There are three sets of candidate craters, and one of the combinations could be a right match for the three craters selected in step 1. For each combination of candidate craters, a linear transform is constructed. Suppose three corresponding craters center are $X_i = (x_i, y_i, 1)$, and $X'_i = (x'_i, y'_i, 1)$, $i = 1, 2, 3$. The transformation T is therefore

$$T = (X_1^T, X_2^T, X_3^T)(X_1'^T, X_2'^T, X_3'^T)^{-1}$$

4. All craters in image 1 are transferred to image 2 by the above transform.
5. All transferred craters are compared with the craters in image 2 in term of the shape and position. Two craters are considered a potential match if their positions are very close (e.g < 3 pixels) and their major and minor axes are also close. For

each combination, the total number of potential matches (n) is counted. If n is greater than a threshold (e.g. 4), all potential matches are used to regenerate another linear transform T by the Least squares and this T will be used to find additional matches by the similar procedure of step 4 and 5.

6. If a match is not confirmed, another combinations will be tested until a match is confirmed or all combinations are exhausted.
7. When the total loops from step 1 to step 6 exceeds a threshold, the matches between the two images are rejected.

Two necessary conditions to ensure the context matching are that there must be enough craters in both images and the relative motion between the two images are known. In addition, because it uses linear transforms to determine matches, all crater have to be close to a plane, which is not always true, particular in the case of a small and irregular body such as Eros.

Conic Invariant Matching

Geometric invariants, which are unaffected by object pose and perspective projection, have been found to be very useful in machine vision. The conic invariants have been studied extensively in the area of machine vision and photogrammetry [5]. A pair of coplanar conics c_1 and c_2 have two invariants and they are

$$\begin{aligned} I_{c_1c_2} &= \text{Trace}(c_1^{-1}c_2) \\ I_{c_2c_1} &= \text{Trace}(c_2^{-1}c_1) \end{aligned} \quad (10)$$

Since under a linear transformation $x = TX$, c_1 and c_2 go to $C_1 = T'c_1T$, and $C_2 = T'c_2T$, then we have

$$\begin{aligned} I_{C_1C_2} &= \text{Trace}(T^{-1}c_1^{-1}(T')^{-1}T'c_2T) \\ &= \text{Trace}(c_1^{-1}c_2) = I_{c_1c_2} \end{aligned} \quad (11)$$

The same derivation holds for $I_{c_2c_1}$.

The procedure for using conic invariants in crater matching is as following:

1. For any two sets of craters (n and m craters respectively) extracted from two images, two hash tables are created and each contains the invariants (Eq. 10) for any permutation of two craters in each dataset. For n craters, the table will be 2 columns by $n(n-1)$ rows, and each row contains $I_{c_1c_2}$ and $I_{c_2c_1}$.
2. An n by m mapping matrix for recording the results of the invariant test is created.
3. Every row between hash tables 1 and 2 are compared. If they are very close to each other, add 1 to the corresponding elements in the mapping matrix.
4. The number of elements in the mapping matrix greater than a threshold (e.g. 4) is counted and if it is greater than another threshold, the matching between the two images can be confirmed otherwise it is rejected. If more than one element in each

row or column have a value greater than a threshold, all associated crater matches will be rejected because of ambiguity.

Like the context based matching, the conic invariant matching relies on sufficient context information to confirm the matches. However, it does not need any motion information, and, therefore, it is more robust than the other two methods. Fig. 6 shows a result of conic invariant matching result of the two images in Fig 3. Identical craters are shown with the same shading. In this case, five identical craters are found.

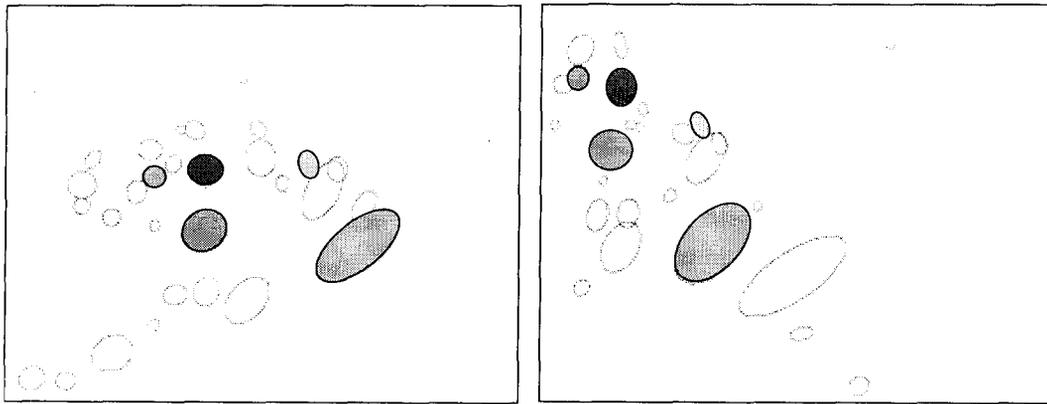


Figure 6 A conic invariant matching result.

The Crater Matching Approach

The three algorithms above have both advantages and disadvantages and they nicely complement each other. The crater matching approach used in the navigation system is a combination of all. The flow chart of this approach is shown in Fig. 7.

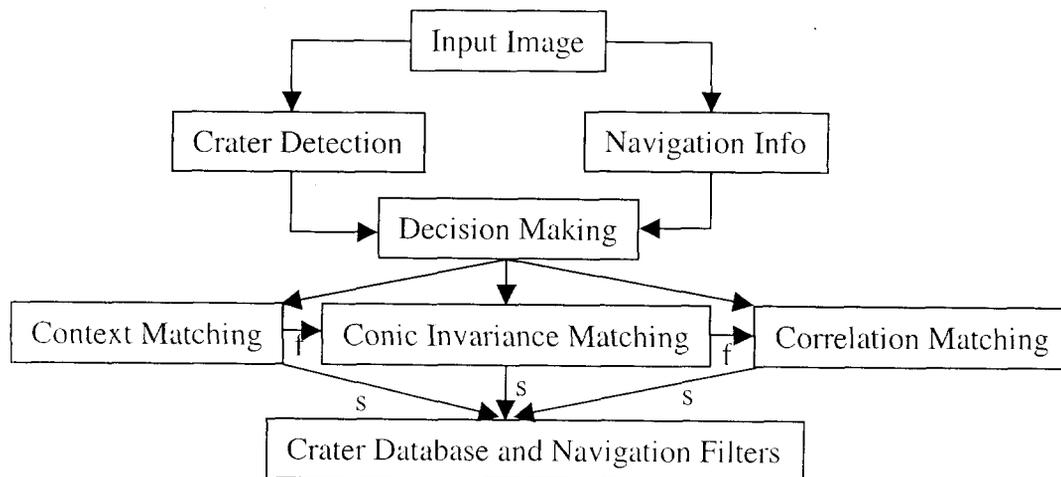


Figure 7: The crater matching is a hybrid approach

As a test of crater matching, two weeks of 200 km NEAR imagery (total 963 images) taken between March 3 and March 23, 2000 were used for a study. A key-influencing factor of photometrical similarity, which decides the feasibility of cross correlation

matching between two images, is the lighting direction. The cross-correlation method is valid if two images cover the same area and have similar lighting angle (e.g. < 10 degrees). Fig. 5 (a) shows the histogram of the photometrically similar images. It indicates that for any image, it is very likely that a similar image covering the same area can be found. Therefore the cross correlation method is a valid method for crater matching. A total of 9288 craters were detected from the 963 images. On the average 9.6 craters were found in each image. Fig. 8b shows the distribution of the number of craters detected from images. Roughly 50% of images have 7 or less craters, which means the less context information is available for context based algorithm. The context matching method is not suitable for these images.

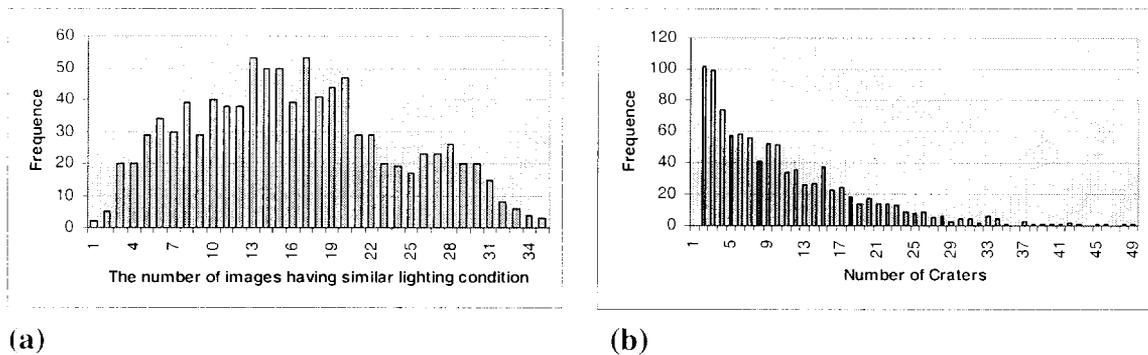


Figure 8: (a) The histograms of the photometrical similarity. (b) Crater frequency.

After the crater matching and the labeling step, 673 identical craters have been positively confirmed in more than one image. These craters are placed in a landmark pool for further orbit determination.

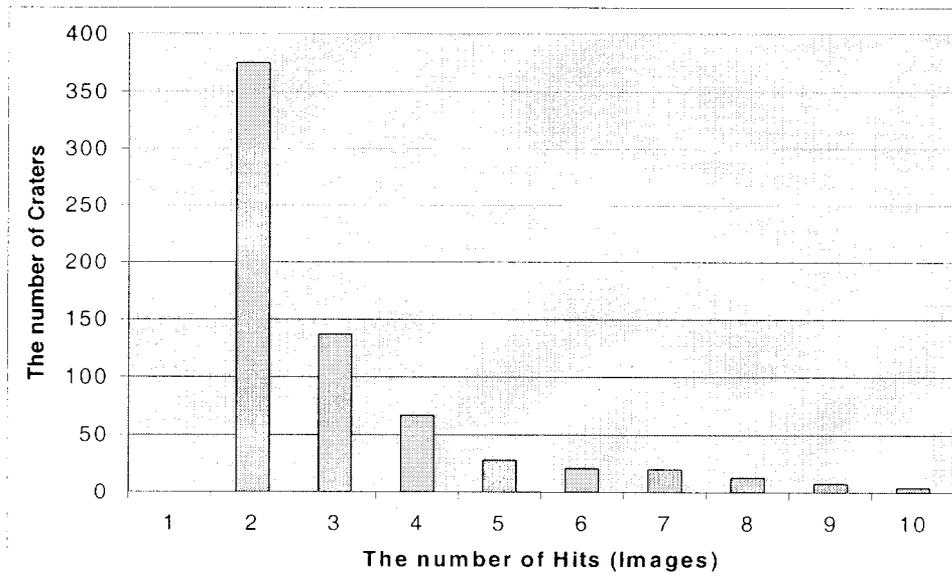


Figure 9: The histogram of the number of hits.

DATA BASE MANAGEMENT

When the crater matching is complete, a file is produced containing image pairs and landmark pairs that are common to the two images. The landmark number can appear on only one image but the same landmark can appear in many different images. Based on the image pairings from the Crater Matching algorithm, a new set of landmark numbers are assigned that are unique to the landmark. Thus the same landmark number may appear on more than one image but cannot appear twice on the same image if the sorting is done properly. The landmark data base management is simply the algorithm for assigning unique landmark numbers. A picture sequence file is then written with a record for each image in time order containing camera pointing and camera parameters and a separate record for each landmark contained on that image containing the line and pixel location. For the first occurrence of a new landmark number, a record is written to a landmark location file that contains the landmark number and it's *a priori* location.

ORBIT DETERMINATION

The picture sequence file and landmark location file are processed along with radiometric data in an orbit determination filter. A new solution for the spacecraft orbit and landmark locations are obtained. If the solution is satisfactory, as indicated by inspection of data residuals, the resulting trajectory solution may be mapped a few days into the future and new images acquired and processed. If the solution is not satisfactory, the entire landmark tracking algorithm may be repeated with the new and presumably better *a priori* spacecraft trajectory.

The system has been successfully tested on the 200 km NEAR orbit imagery. In order to evaluate its performance, numerous scenarios were designed to simulate the worst to the best real mission situations. First, the order of the spherical harmonic shape model was varied from a coarse second-degree model to a high precision model of degree 34. Second, the crater matching threshold was varied from a low probability of matching that intentionally introduced a certain number of outliers to a high probability of matching that virtually guaranteed no mismatched craters. The system has been able to detect and reject the outliers and converge to correct solutions in all test cases. This experimental study showed that the OD accuracy of this system is competitive with the manual approach and is at least 2 orders of magnitude faster. In the 200km orbit, 100 m accuracy was obtained after an hour of processing, which required several weeks of processing using manual crater detection and matching (Fig. 10).

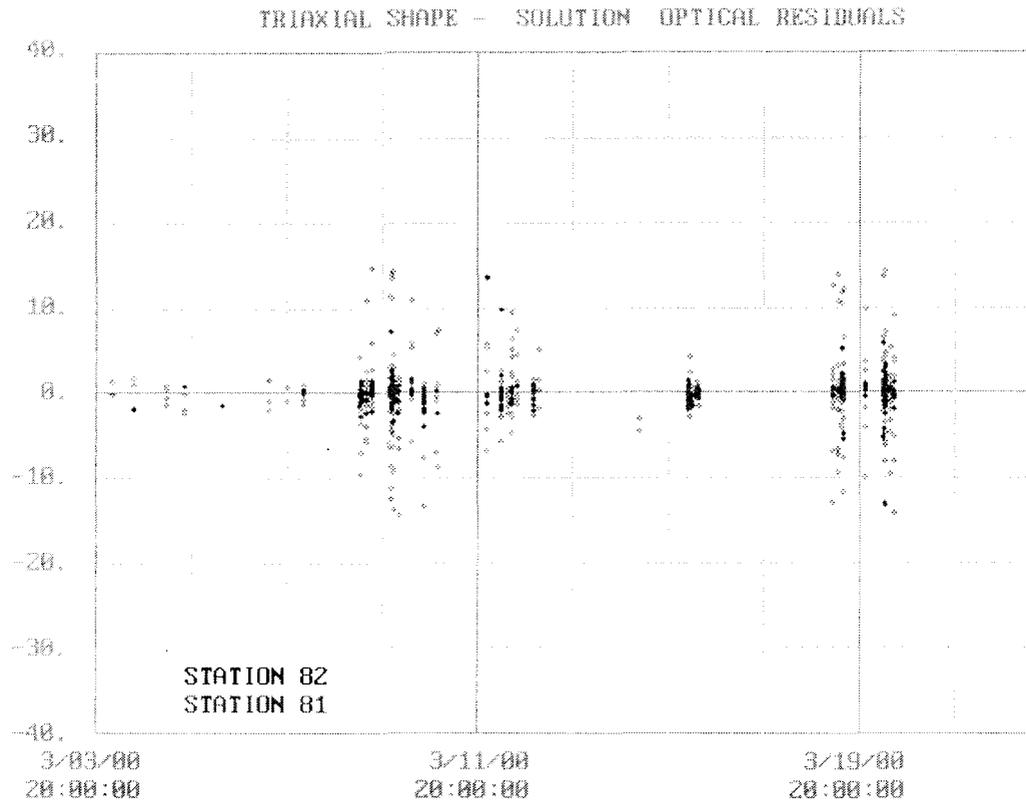


Figure 10: Optical data residuals in Pixels.

CONCLUSIONS

In this paper an autonomous landmark based spacecraft navigation system is discussed. Several key innovations have made this approach possible. The image selection and planning scheme enables selection of an image sequence and other navigation data from a huge data set. The crater detection algorithm is able to detect craters precisely and robustly. It does an excellent job of detecting craters from NEAR imagery, which has been of poor quality due to fuel leakage. The crater matching takes the advantages of three different algorithms to achieve very robust crater identification. Finally, the outlier rejections are able to detect any outliers and make the convergence faster and reliable.

Future work in this area will concentrate on an extensive test of lower orbit imagery (< 100 km). There will be some difficulties in the lower orbit case because more images and more craters will be involved which will be a tough test of the robustness of all procedures in the system. If it passes this test, it will be ready for the future planetary body exploration missions, such as the Dawn (2006) and Mercury Message (2009) missions.

Development of a fully onboard autonomous spacecraft navigation system is a goal that is many years in the future. An autonomous navigation system would process navigation data acquired by the spacecraft, determine the spacecraft position and

velocity, design a trajectory to acquire certain target parameters, execute maneuvers to attain the target and use the spacecraft position to control science instrument pointing. The algorithms required to perform autonomous navigation are essentially the same as those that are discussed in this paper. The essential difference is the use of a computer on board the spacecraft rather than ground-based computers. In the past, the development of autonomous navigation has been hindered by the capability of computers on board spacecraft. The speed, memory and storage requirements were inadequate for full autonomy. Once these restrictions are removed, it will open a new window for many applications, such as pinpoint landing on Mars (abundant craters) or asteroids or autonomous onboard navigation etc.

ACKNOWLEDGEMENTS

This paper describes work performed at the Jet Propulsion Laboratory, California Institute of Technology, under contract from 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. J. K. Miller et al "Determination of Shape, Gravity, and Rotational State of Asteroid 433 Eros", *Icarus* 155 3-17 (2002).
3. John Canny, "A computational approach to edge detection," *IEEE PAMI*, 8 (6) 1986.
4. Marina Kolesnik and Gregory Baratoff, "3-D interpretation of sewer circular structures", *Proceedings of 2000 IEEE ICRA*.
5. David Forsyth et al, "Invariant descriptors for 3-D object recognition and pose" *IEEE PAMI*, Vol. 13, No. 10, 1991.
6. Long Quan, "Invariant of a Pair of Non-coplanar conic in space: definition, geometric interpretation and computation, *IEEE CVPR*, 1995.
7. Yves Dufournaud, C. Schmid, R. Horaud, "Matching Image with Different Resolution", *IEEE CVPR*, 2000.