

LUNAR CRATER IDENTIFICATION USING TRIANGLE REPROJECTION

Ava C. Thrasher*, John A. Christian†, Giovanni Molina‡, Michael Hansen§, John Y. Pelgrift¶ and Derek S. Nelson||

Image-based terrain relative navigation is a critical capability for future lunar exploration missions. Images of the lunar surface containing craters can be compared to on-board maps to identify craters and estimate the spacecraft position. While there are many ways to accomplish the crater identification task, this work explores a method using triangulation and crater triangle pattern projections. Specifically, potential matching crater patterns from the catalog and image are used to triangulate the spacecraft position, allowing for construction of line-of-sight directions to the potential matching catalog craters. The projection of these directions in the image can be compared to the observed craters to accept or reject the match hypothesis. In this paper, we demonstrate the algorithm's capability in handling various types of input errors and what tolerances can be tuned to achieve a desired performance. Additionally, an initial look at flight software implementation is included.

INTRODUCTION

New lunar exploration missions with challenging navigation requirements have created a need for new terrain relative navigation (TRN) algorithms. Recent work in this area has considered a variety of approaches, including visual odometry¹ and landmark matching (e.g., computer vision features,^{2,3} SPC maplets,⁴ craters^{5,6}). In this work we consider the problem of crater-based TRN, focusing specifically on the task of autonomous crater identification.

Our work in this area is motivated by the needs of the Intuitive Machines' Nova-C lander (Fig. 1), which we anticipate will require crater-based navigation capability beginning with the IM-2 mission planned for 2024. The IM-2 mission will be Intuitive Machines second lunar payload vehicle. One of IM-2's primary payloads will be PRIME-1, whose objective is to search for water, CO₂, and other compounds on the Moon. NASA, the owner of PRIME-1, has identified areas of high probability for water at the Moon's south pole. The desired landing sites are treacherous, with only narrow regions where it is possible for a vehicle to land and perform operations. Because of this, IM-2 must land within 50 meters of its intended landing site (ILS). To achieve this precise landing requirement, IM-2 will have two measurements to observe its position relative to the terrain: line-of-sight (LOS) measurements to crater centers and a LIDAR-based terrain matching measurement. The resulting measurements will be passed to the vehicle's onboard filter for real-time state estimation.

*Graduate Research Assistant, Guggenheim School of Aerospace Engineering, Georgia Institute of Technology, Atlanta, GA 30332.

†Associate Professor, Guggenheim School of Aerospace Engineering, Georgia Institute of Technology, Atlanta, GA 30332.

‡Software Engineer, Intuitive Machines, Houston, TX 77058.

§Nova-C Navigation Lead, Intuitive Machines, Houston, TX 77058.

¶Optical Navigation Software Development Lead, KinetX SNAFD, Simi Valley, CA 93065

||Optical Navigation Engineer, KinetX SNAFD, Simi Valley, CA 93065



Figure 1 Intuitive Machines Nova-C lunar lander.

At a high-level, most image-based crater navigation schemes are comprised of three parts: crater detection, crater identification, and state estimation. When provided an image, the first task is to apply image processing techniques to extract the apparent location (and possibly extent) of craters in the image. This is accomplished by a crater detection algorithm (CDA). The second task is to associate each of the observed craters with known craters (e.g., in a catalog, in another image) or discard them as false returns. This is accomplished by a crater identification algorithm (CIA). Having now associated image plane points with three-dimensional (3-D) world points, the third task is to use the matched observations for state estimation. Although instantaneous localization may generally be achieved by triangulation, the actual state estimation usually occurs within an onboard navigation filter. While this work will touch on all three aspects of the crater-based navigation architecture for IM-2 to some extent, our focus in this manuscript is on the crater identification component of this system.

The IM-2 CIA was developed to match craters found in a digital image to a catalog of known lunar craters. While some crater identification methods solve the (nearly) global lost-in-space problem with invariants,^{6,7} we do not attempt to solve such a generic problem here due to the nature of the IM-2 navigation architecture. Instead, this work focuses on identifying craters along a planned trajectory on which the spacecraft always has a reasonable onboard state estimate. Good *a priori* state knowledge allows us to anticipate the apparent geometry of crater patterns to make the matching problem much simpler, as illustrated in Fig. 2. Our approach generates a list of potential crater matches for a triad in the catalog using the triangles' interior angles. These match hypotheses may be used to estimate the spacecraft location by triangulation,⁸ and are then validated by synthetically reprojecting the catalog craters into the image and comparing them to the original observations.

BACKGROUND

Camera Model and Projective Geometry

We assume the use of a conventional camera which forms images according to perspective projection. The geometry of perspective projection is shown in Fig. 3, where the so-called *pinhole camera model* relates a line-of-sight (LOS) direction to points on the focal plane (inside the camera) and the image plane (outside the camera). The pinhole location is an idealized point along the optical axis through which a ray of light may pass without changing direction. For this and many other applications, it is used as the origin of the camera frame. We choose a convention that places

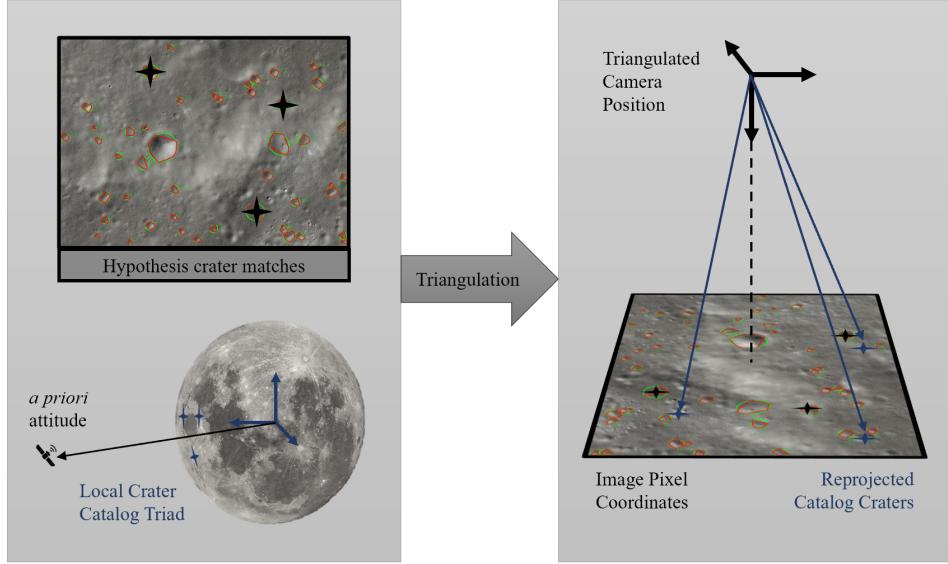


Figure 2 The crater ID algorithm uses triangulation to compare image crater triads with catalog crater triads as potential matches.

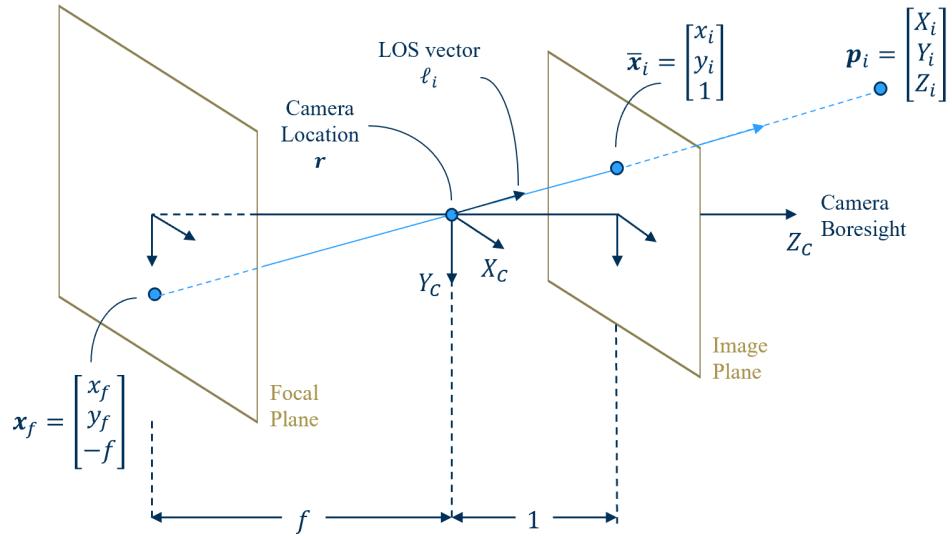


Figure 3 This mathematical model of a projective camera allows for transformation from a point in the world frame to a point in the image.

the camera frame's positive Z_C axis along the camera boresight (out of the camera), positive X_C axis to be in the direction of increasing column count in the digital image, and positive Y_C axis to complete the right-hand set.⁹

The image plane—a mathematical (nonphysical) construct defined as the $Z_C = 1$ plane—has a convenient placement that allows for efficient operations in projective space. Suppose that we have a LOS direction, denoted by the line $\ell_i \in \mathbb{P}^2$, that passes through the camera location $r \in \mathbb{R}^3$ and an observed point on the lunar surface $p_i \in \mathbb{R}^3$. Further define the point where this line pierces

the image plane as $\bar{\mathbf{x}}_i^T = [x_i, y_i, 1]$. This pinhole camera geometry relates these points through a proportionality relationship

$$\ell_i \propto \bar{\mathbf{x}}_i \propto \mathbf{p}_i - \mathbf{r} \quad (1)$$

Transformation between homogeneous image pixel coordinates $\bar{\mathbf{u}}^T = [u, v, 1]$ and the image plane coordinates $\bar{\mathbf{x}}$ is described by an affine transformation

$$\bar{\mathbf{u}} = \mathbf{K} \bar{\mathbf{x}} \quad (2)$$

where the 3×3 matrix \mathbf{K} is the so-called camera calibration matrix. See Ref. 9 for more details on the camera calibration matrix and how it relates to specific camera parameters.

Image Processing and Crater Detection

Although this manuscript primarily focuses on the crater identification problem, we find it helpful to briefly review the CDA that produces the inputs to the CIA. The current CDA works by identifying areas of shadow and then pairing them with adjacent bright regions. For every shadow region identified, a search area of proportional size is created in the direction of the Sun's illumination in the image. If a bright region is within the search area, the shadow and light region are associated with one another. Additional processing is used to create one contour for each shadow-bright region pair, and then an ellipse is fit to the contour. The ellipse is then defined as a potential crater if the total area of the shadow and bright regions are greater than a specified percentage of the ellipse and if the ellipse meets a prescribed ellipticity range. If the ellipse is determined to be a crater, then the center of the ellipse is defined as the crater location. Due to the stability and repeatability of the CDA-produced crater rim descriptions, the CIA only uses the crater center locations within the matching procedure.

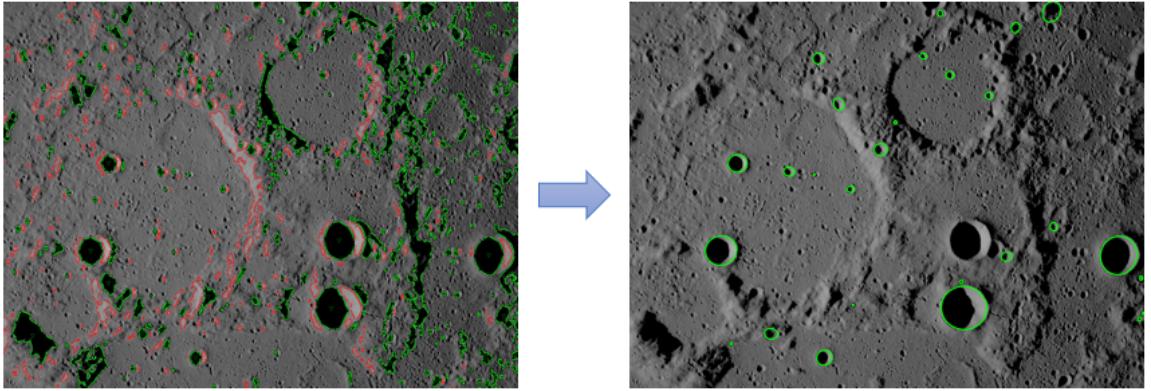


Figure 4 Demonstration of CDA on a synthetic image of the Moon. The algorithm begins by finding dark (green outline) and light (red outline) regions of the image (left frame). These regions are grouped and used to create elliptical fits of candidate crater rims (right frame).

Triangulation Techniques

Triangulation is a classical method for localizing a point as the intersection of LOS directions. These ideas significantly predate space travel,⁸ but continue to be a powerful and practical tool.

Triangulation may take the form of resection or intersection, as shown in Fig. 5. Resection is the triangulation method which can determine a point position by observing at least two other points at known locations, while intersection determines the position of an unknown point by observing it from at least two other known locations.⁸ Navigation by simultaneous LOS observations to a set of craters is a resection problem.

There are various methods to solve triangulation problems regardless of the problem being resection or intersection. For the purposes of triangulation within the crater ID pipeline, we do not need a statistically optimal solution—but rather a reasonably good, but exceptionally fast, solution. Therefore, we choose to accomplish fast, suboptimal triangulation via the classical direct linear transform (DLT). If one wishes to obtain a position fix for direct use within a navigation filter, an optimal triangulation method (e.g., LOST⁸) would be more appropriate.

The development of the DLT algorithm is straightforward. Begin by considering a spacecraft at position \mathbf{r} that obtains $n \geq 2$ simultaneous LOS measurements $\{\ell_i\}_{i=1}^n$ of objects (e.g., craters) at known position $\{\mathbf{p}_i\}_{i=1}^n$. Since the LOS direction measurement contains no range information, we find ℓ_i to be of arbitrary scale as indicated by the proportionality relation in Eq. (1).

To eliminate the unknown scale, take the cross product of both sides with the LOS measurement

$$[\ell_i \times] \ell_i \propto [\ell_i \times] (\mathbf{p}_i - \mathbf{r}) = \mathbf{0}_{3 \times 1} \quad (3)$$

which simplifies to

$$[\ell_i \times] \mathbf{r} = [\ell_i \times] \mathbf{p}_i \quad (4)$$

Here the cross matrix indicated by $[a \times]$ is equivalent to taking the cross product of two vectors such that

$$[a \times] b = a \times b \quad (5)$$

The result from Eq. (4) is valid in any consistent frame. For this application, the measurement LOS vectors ℓ_i will be acquired in the camera frame (C) and the crater positions \mathbf{p}_i will be stored onboard in a lunar centered lunar fixed (LCLF) frame (L). Thus, we may use the rotation matrix

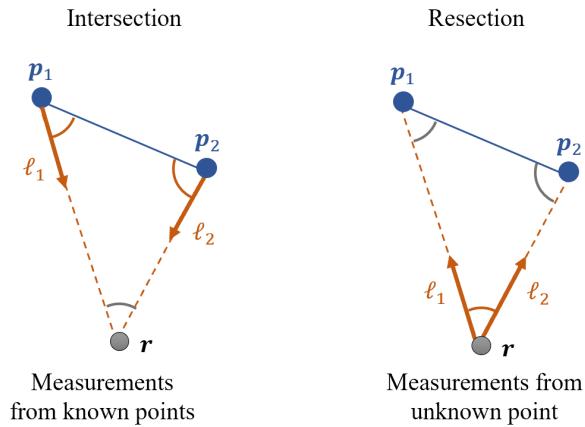


Figure 5 Triangulation can be done with measurements from points of known location (intersection) or from the point of unknown location (resection). Use of LOS observations for triangulation is a resection problem.

\mathbf{T}_C^L (describing a transformation from L to C) to bring the LCLF crater positions into the camera frame.

Therefore, rewriting Eq. (4) with ℓ_i and \mathbf{p}_i expressed in their native frames, one arrives at

$$[\ell_{C_i} \times] \mathbf{T}_C^L \mathbf{r}_L = [\ell_{C_i} \times] \mathbf{T}_C^L \mathbf{p}_{L_i} \quad (6)$$

where we have chosen to express the camera's position in the LCLF frame, \mathbf{r}_L . Given $n \geq 2$ LOS measurements corresponding to known points, Eq. (6) can be stacked to form a linear system that allows for a direct solution of the camera's location at \mathbf{r}_L

$$\begin{bmatrix} [\ell_{C_1} \times] \mathbf{T}_C^L \\ [\ell_{C_2} \times] \mathbf{T}_C^L \\ \vdots \\ [\ell_{C_n} \times] \mathbf{T}_C^L \end{bmatrix} \mathbf{r}_L = \begin{bmatrix} [\ell_{C_1} \times] \mathbf{T}_C^L \mathbf{p}_{L_1} \\ [\ell_{C_2} \times] \mathbf{T}_C^L \mathbf{p}_{L_2} \\ \vdots \\ [\ell_{C_n} \times] \mathbf{T}_C^L \mathbf{p}_{L_n} \end{bmatrix} \quad (7)$$

CRATER IDENTIFICATION METHODOLOGY

The crater identification pipeline is summarized in the flowchart depicted in Fig. 6. Recognizing that the current CDA provides only the apparent crater center coordinates (and not rims), the fundamental idea behind our matching scheme is to compare triangles formed by the image centers of various crater triplets. The comparison is performed between the observed crater centers (i.e., output from CDA) and what is expected from reprojection of the crater catalog.

Suppose we are provided an image of the lunar surface and the CDA detects the existence of n potential craters. Let the image coordinates of these n potential craters be denoted by $\{\bar{\mathbf{u}}_i\}_{i=1}^n$. From here, we may find the $\binom{n}{3}$ unique triangles formed by all possible combinations of the potential craters. Assuming no three co-linear crater centers, these $\binom{n}{3}$ triangles of observed craters may be stored in a database containing the triangles' interior angles and corresponding indices. To address permutation issues, we choose to store these triangles (both angles and indices) within the database in a clockwise fashion with the largest interior angle appearing first.

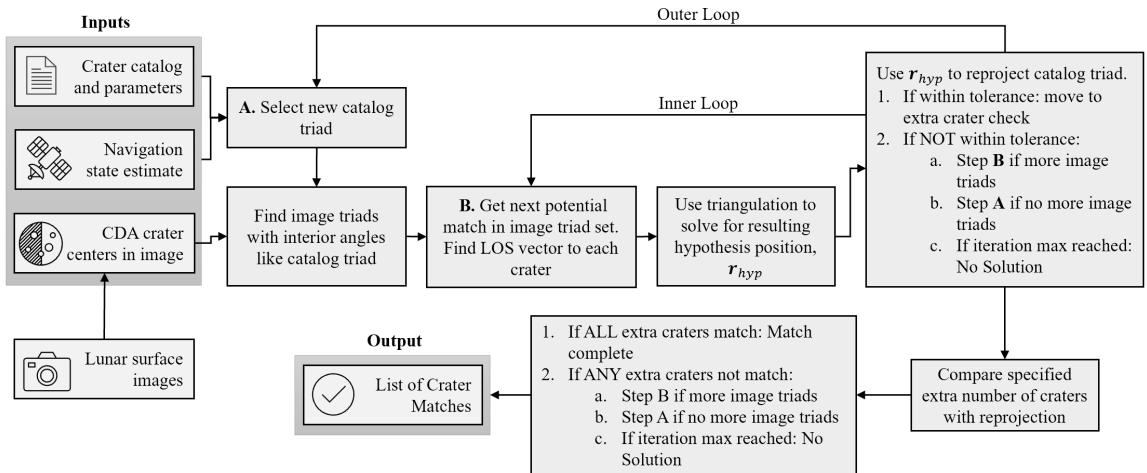


Figure 6 The crater identification scheme queries through crater triads in the catalog and checks for a match within a selected group of image crater triads.

Having computed all possible triangles from craters within the image, we must now consider how to match this geometry against a catalog of known 3-D crater locations. Because we are not using view invariants¹⁰—in this case we are using image triangle interior angles that change under perspective projection with different camera viewpoints—a database of catalog interior angles cannot be constructed ahead of time. Instead, the interior angles must be computed on-the-fly. To achieve this, we use *a priori* state knowledge to query our onboard catalog (more on this in a different section of this paper) for crater centers that should be within the camera’s field of view (FOV). We expand our query beyond the actual FOV to account for onboard state uncertainty. The catalog craters returned by this query are then projected into expected image points using our *a priori* state knowledge of \mathbf{r}_L and \mathbf{T}_C^L .

Our matching algorithm then proceeds by attempting to match triads from the catalog to the database of triads from the image. Matching the catalog to the image (instead of the image to the catalog) may seem backwards, but this scheme was chosen intentionally. Since the database of triangles and their interior angles may be computed from either the catalog or image, our objective is to minimize (on average) the number of triads we must evaluate. This was achieved by choosing to form the database out of the smaller of the two lists, which we posit will usually be the list of observed image craters. This design choice may be revisited later.

The method for querying catalog triads should be chosen strategically so that the algorithm shuffles through comparisons in a way that reduces sequential trials with common craters. This helps us avoid dwelling on a crater that doesn’t exist in the database. Such a shuffling scheme decreases the time (on average) to stumble upon a triplet that exists in both the image and catalog. We achieve this here using enhanced pattern shifting (EPS),¹¹ but other suitable methods also exist.

The procedure for finding potential triad matches from the image database for an EPS-generated query triad (drawn from the catalog) is mechanized as follows. The three interior angles of the candidate triad (as projected into a synthetic image with *a priori* state knowledge) are computed, stored in clockwise order, and cyclically permuted to bring the largest angle into the first position. We then find the triads in the database (as formed from the potential craters found in the image) whose largest angle is within a specified tolerance of the query angle. Then, amongst those returns, we find the triads whose intermediate angle is also within the specified tolerance. It is possible to include other metrics as well (e.g., crater radius^{5,12}), but this is not done here. The result of the database query process is a (hopefully short) list of image triads whose interior angles are within a specified tolerance of the catalog query triad. Each of these represents a match hypothesis that must be tested.

With a short list of match hypotheses in hand, the algorithm will then proceed as follows. Using the catalog LCLF positions $\{\mathbf{p}_i, \mathbf{p}_j, \mathbf{p}_k\}$ for the EPS-selected query triad and the LOS directions for the currently hypothesized image craters $\{\ell_n, \ell_m, \ell_o\}$, we may perform the DLT using Eq. (7) to generate a hypothesized camera location $\mathbf{r}_{L_{hyp}}$. This is illustrated in Fig. 7. Then, using this hypothesized camera location, we reproject the LCLF crater positions back into the image as

$$\boldsymbol{\ell}_{C_{hyp,i,j,k}} \propto \mathbf{T}_C^L (\mathbf{p}_{L_{i,j,k}} - \mathbf{r}_{L_{hyp}}) \quad (8)$$

which may be written explicitly in focal plane coordinates

$$\bar{\mathbf{x}}_{hyp,i,j,k} = \frac{\boldsymbol{\ell}_{C_{hyp,i,j,k}}}{\mathbf{k}^T \boldsymbol{\ell}_{C_{hyp,i,j,k}}} \quad (9)$$

where $\mathbf{k}^T = [0, 0, 1]$. These may, in turn, be converted to pixel coordinates by rearranging Eq. (2) to find

$$\bar{\mathbf{u}}_{hyp,i,j,k} = \mathbf{K}^{-1}\bar{\mathbf{x}}_{hyp,i,j,k} \quad (10)$$

The hypothesized crater reprojections $\bar{\mathbf{u}}_{hyp,i,j,k}$ are then compared to the measured crater centers $\bar{\mathbf{u}}_{n,m,o}$. If they agree to within a specified tolerance, we proceed to the next step. If not, then that hypothesis is rejected and the next hypothesis image triad is pulled to test against the current catalog triad. If the list of hypothesis image triads is exhausted with no match, then a new catalog triad is queried (next combination in the EPS sequence) and the matching process is repeated.

If the reprojected craters are found to be within tolerance, then a set of extra craters should be checked to confirm the match and reduce the likelihood of a false positive. After the original triad is matched in this algorithm, the remaining craters in the catalog are reprojected using the same hypothesized camera position. A minimum number of these craters must also have a match in the image for the triangle to be verified. Should this additional check meet the set tolerance, then the triads as well as the additional craters are marked as a match.

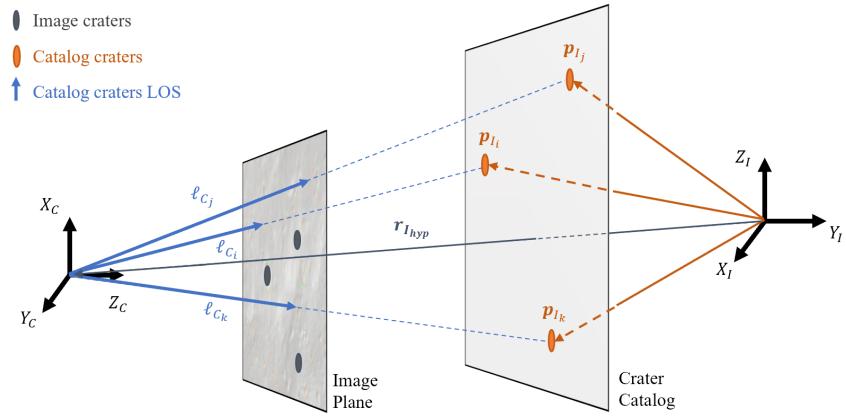


Figure 7 Once the hypothesis camera position has been determined from the potential match in the image, the catalog crater positions are reprojected.

IMAGE AND CATALOG CREATION

There are at least two reasonable ways to construct a crater catalog for navigation at the Moon. The first method is to repurpose one of the many scientific crater catalogs. The second method is to build a custom catalog informed by the same CDA to be used during flight. Both have their advantages and are useful in different phases of development, testing, or operations.

Scientific crater catalogs are most helpful during the development and testing phases. They provide a quick and easy means of obtaining representative crater distributions and shapes, that are helpful in planning trajectories or developing algorithms. Although a great many crater catalogs exist, we primarily use the so-called Robbins Crater Catalog.¹³ This catalog contains over 2-million craters on the lunar surface created primarily using data from the Lunar Reconnaissance Orbiter (LRO) and SELENE (or “Kaguya”). Although the Robbins catalog contains information about the craters’ size, ellipticity, position, we only require the craters’ center position and ID for the IM-2 CIA algorithm development. Using a reference trajectory as truth, the full Robbins catalog was

reduced from global coverage to only craters within a conservative range around the spacecraft’s reference trajectory. The resulting catalog can be used to generate synthetic images by analytic projection of crater centers, which permits a controlled environment for interrogating the effects of different error sources (see Fig. 8).

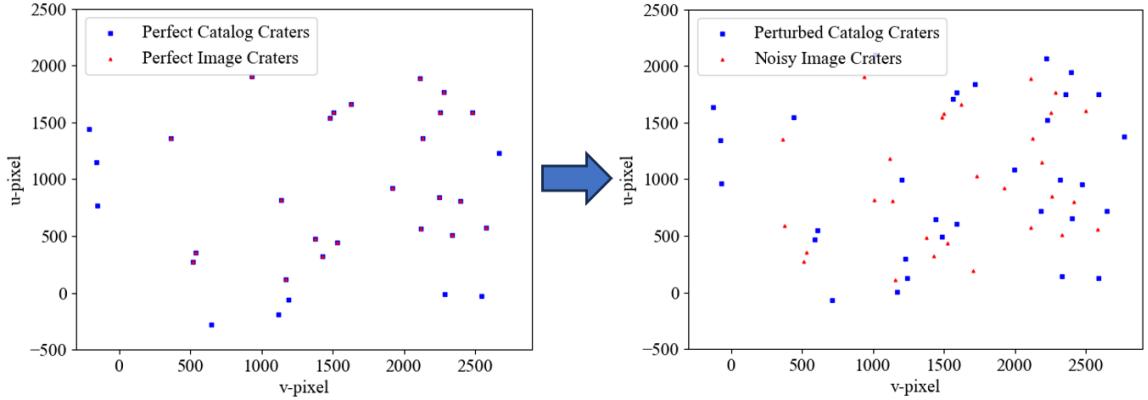


Figure 8 Robbin’s crater database was used to form a catalog and synthetic image data which can be perturbed with state and/or image processing errors.

An alternative to scientific crater catalogs is to build a custom catalog with the same CDA that will be used to perform TRN in flight. Using a CDA to generate a catalog with photorealistic synthetic imagery (or on images from past missions) allows us to populate the catalog with the type of craters the CDA will observe in flight—instead of the type of craters deemed important for scientific study. The downside of this approach is that it is expensive to construct such a custom catalog for every descent trajectory, and many flight-appropriate CDAs produce lower accuracy centroids than available from legacy catalogs.

RESULTS

Crater Identification Performance

Understanding the effect of different error sources in the CIA inputs (e.g. *a priori* state knowledge and CDA crater center errors) motivates a series of Monte Carlo numerical experiments. Of particular interest is the study of how different CIA parameters affect overall algorithm performance. For these experiments, a particular spacecraft state was chosen along the trajectory, and crater locations were pulled from the Robbins crater catalog.

The 1σ error levels for the spacecraft state (shown in Table 1) were chosen to be stress the algorithm (significantly larger than expected in flight), while the pixel error applied to the image craters was selected to approximate worst-case CDA performance. To test state error effects, the synthetic image was generated using the camera state pulled from the trajectory (the truth state), while the crater catalog segment was selected with the perturbed state (imperfect *a priori* knowledge). To test the effects of imperfect crater detection, the synthetic image crater centers were perturbed. Additionally, extra craters returns were randomly placed in the image with a uniform distribution, such that 20% of the image craters were not in the catalog (and, thus, have no match to be found). This simulates instances of the CDA detecting craters (or other surface features) not included in the catalog. For all runs, the image and catalog orders were shuffled to randomize the query order through

the catalog list.

Table 1 Crater identification Monte Carlo errors. Values indicate standard deviation (1σ) for a Gaussian distribution.

Parameter	Value
Position Error	2.0 km
Attitude Error	0.5 degrees
Image Pixel Error	8 pixels

This CIA has three main tolerances that can be adjusted to achieve desired performance. The pixel tolerance dictates the radius around an image crater in which a projected catalog crater must fall to be considered a match. The interior angle threshold is used when searching for potential matches from the database of image crater triads. Finally, a specified number of extra craters can be checked after the initial triad is matched to verify the result. Initially, these parameters were set somewhat arbitrarily, informed only by the predetermined error levels. The pixel tolerance was chosen as the 3σ value for pixel error (24 pixels) and the interior angle threshold was set to be 0.005 radians. The extra number of craters to verify was varied from 0 to 2 to analyze the effects.

The CIA accuracy results using these unrefined tolerances are shown in Table 2. Results are shown for Monte Carlo analyses with individual error types (pixel, position, or velocity) and with all the error types together. The rate of false positives was unacceptable for our application, and indicated further adjustment of the tolerances was necessary. However, useful information was discovered from this initial investigation. Of the three error types in this test case, pixel error clearly had the greatest affect on the accuracy and efficiency of the algorithm—this underscores the importance of a reliable and repeatable CDA. Increasing the number of extra craters verified from 0 to 2 improved performance, but proved to not be enough to completely reject false positive cases.

Table 2 Crater identification accuracy results revealed that the choice of uninformed tolerances were unable to reject all false positives in the case of all errors applied.

Ex. Craters	All Error			Pixel Error			Position Error			Attitude Error		
	0	1	2	0	1	2	0	1	2	0	1	2
True	90.6	97.2	98.7	91.3	98.2	99.1	100.0	100.0	100.0	100.0	100.0	100.0
False	9.4	2.8	1.3	8.7	1.8	0.9	0.0	0.0	0.0	0.0	0.0	0.0
None	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

To eliminate false positives in the crater ID results, the tolerances were tightened and their resulting effects were observed. By reducing the interior angle threshold at which image triads were selected as potential matches, the list was shortened to only triads with extremely similar geometry. While this is reduced the risk of false positives, it also increases the possibility of not including the true match in the selection of potential matches. Tightening the pixel tolerance removed cases in which a false image match was perturbed enough such that the catalog projections would fall within the matching tolerance. However, there remained some cases in which craters in the image were already close together without any perturbations. To address this issue, a new tuning parameter was introduced: a minimum distance requirement between craters in the image. By setting a minimum distance between any two craters in the image, the likelihood of two craters being close enough to

match to the same catalog crater was greatly reduced. The adjusted parameters—which were able to reject all cases of false positives when verifying with extra craters—are shown in Table 3.

Table 3 Initial tolerances used for crater identification were not sufficient in false positive rejection, so the parameters were tuned until all false positives were rejected.

	Pre-tuning	Post-tuning
crater distance (pixels)	None	75
pixel tolerance	24	8
interior angle (radians)	0.005	0.001

The combination of these new tolerances produced no false positives when two additional craters were checked, as shown in Table 4. However, this came at the expense of a small percentage of cases in which no match was found. Additionally, these new results from the tuned parameters confirmed the original observation that pixel error had the greatest affect on accuracy in these experiments. While error in only the spacecraft state yielded only true matches (despite the number of extra craters checked), the pixel error alone resulted in false positives. This result is crucial, because it indicates that accurate detection of crater centers in the image (i.e. the output of a CDA) is paramount to the performance of this crater ID algorithm. The clear importance of requiring resilience to crater detection error led to an investigation on what types of triangles are best suited for this application, which will be discussed in the next section. The increased accuracy of the algorithm also resulted in a higher average catalog queries needed, as shown in Fig. 9. This is expected because as tolerances are tightened and false positives are rejected, it becomes more likely that a perturbed true positive match will not meet the tolerances.

Table 4 Crater identification accuracy results using tuned tolerances.

Ex. Craters	All Error			Pixel Error			Position Error			Attitude Error		
	0	1	2	0	1	2	0	1	2	0	1	2
True	96.5	99.7	99.3	98.3	99.9	100.0	100.0	100.0	100.0	100.0	100.0	100.0
False	3.5	0.2	0.0	1.7	0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0
None	0.0	0.1	0.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

The triangulated spacecraft positions associated with crater matches produced by the CIA were also recorded for the tuned tolerance Monte Carlo run when two craters were checked. The residuals of these estimated positions were calculated and their frequencies plotted (in Fig. 10) in the orthogonal cartesian world frame planes (i.e. spacecraft position error in the x and y direction are plotted in the xy-plane). These residuals were found to be primarily zero mean. These results are useful as a baseline for comparison to flight software operation, which is discussed later.

Triad Geometry Sensitivity

This CIA matches observed crater triads to catalog crater triads using triangle interior angles. Therefore understanding which triangle geometries are resilient to error (particularly to pixel error) can be useful for determining which cases have the highest likelihood of returning a false match (and so should be avoided).

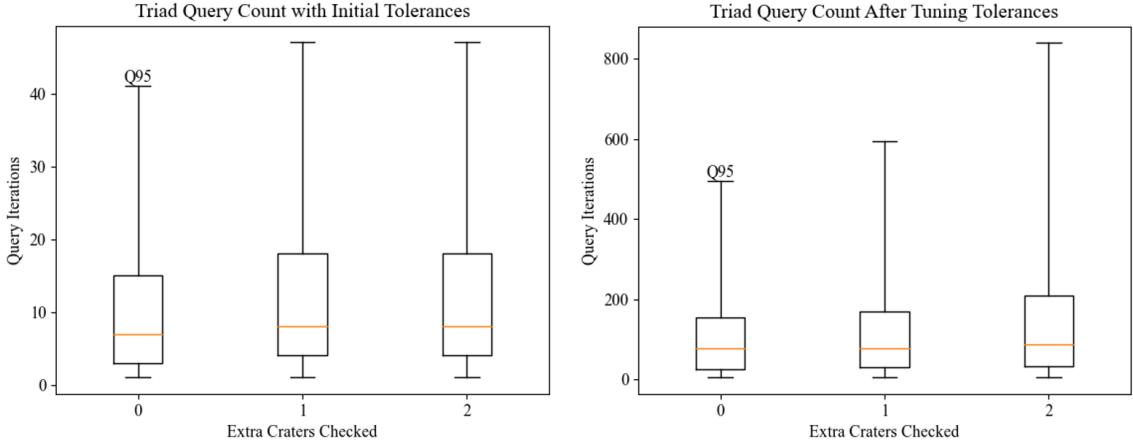


Figure 9 The average number of queried triad checks until a match was found increased after tuning the tolerances to reject false positive crater matches, and the distribution skewed due to cases when no match was found (Marked quartiles are 5, 25, 50, 75, and 95).

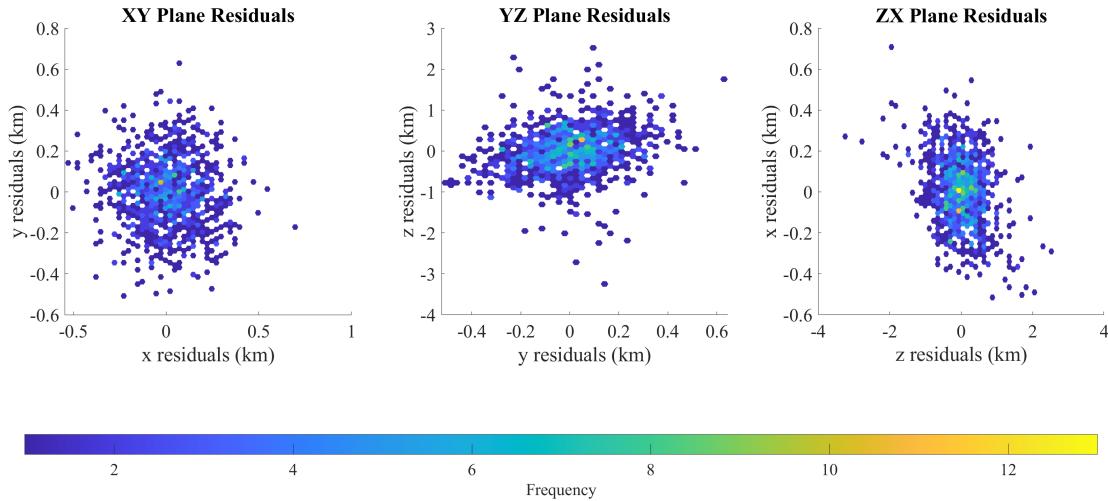


Figure 10 The triangulated spacecraft positions calculated during the tuned tolerances Monte Carlo experiment were compared against the true spacecraft state and the frequency of residual is shown to be near zero mean in all three world frame axes.

A Monte Carlo analysis was performed to determine which geometries are the most resilient (i.e., which geometries have the most stable interior angles). To obtain salient results, we simplify the triangle parameterization. Specifically, without loss of generalization, we normalize the longest leg of the triangle to unity—placing one triangle vertex at $(0,0)$ and another triangle vertex at $(0,1)$. The third triangle vertex must simultaneously lie within both a unit circle centered at $(0,0)$ and at $(0,1)$ since the legs to the third vertex must both be less than unity by construction. Every valid unique triangle geometry was randomly perturbed with a 1σ error of 5% of the longest leg for a Monte Carlo of size 1000. These noisy points were used to calculate the resulting interior angles. The heat

maps in Fig. 11 show the 1σ error of the calculated interior angles as a function of the placement of the third vertex. The Monte Carlo results indicate that isosceles triangles will be the most robust against error. However, isosceles triangles are difficult to match in the CDA because all three of the interior angles are the same. Conversely, triangles which have extreme geometry—such as one leg approaching the length of the hypotenuse—are greatly affected by perturbing the vertex placement. These triangles should therefore be avoided during crater identification.

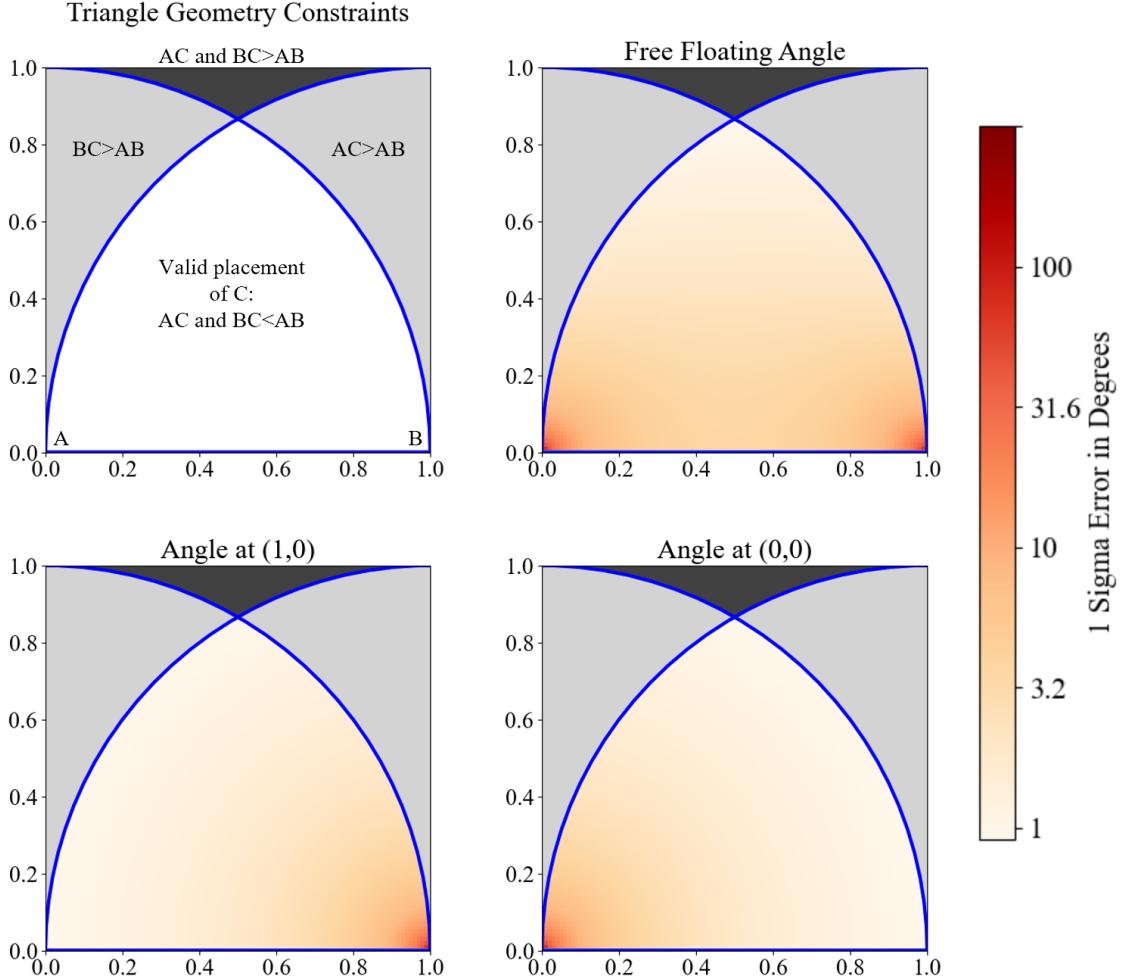


Figure 11 Triangle geometry sensitivity results from Monte Carlo study reveal that isosceles triangles with all legs of near equal length are most resilient to random error. These triangles, however, are hard to distinguish from one another and have an ambiguity in cyclic permutation of the vertex indices.

FLIGHT SOFTWARE IMPLEMENTATION AND TESTING

The CDA and CIA are currently being implemented and tested within the flight software of the Intuitive Machines Nova-C lander for the IM-2 mission. The CDA processes images taken by one of the two optical navigation imagers onboard the spacecraft, and the CIA identifies those detected craters, using the current estimate of the spacecraft position and attitude from the onboard navigation

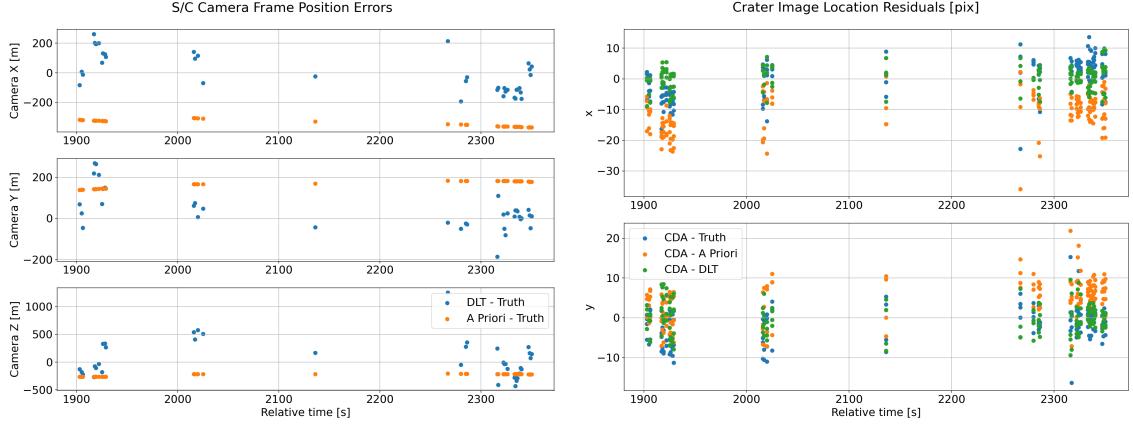


Figure 12 Sample preliminary results from the flight software implementation of the CIA. Left: DLT-estimated and *a priori* spacecraft position errors with respect to the true simulated positions. Right: Residuals between the observed crater image locations determined by the crater detection algorithm (CDA) and crater image locations computed using the DLT, *a priori* and truth spacecraft positions.

filter for its *a priori*. The flight software is tested using full software-in-the-loop simulations of the planned lunar orbit and landing trajectory with perturbations. The images input to the CDA are simulated using Unreal Engine and digital elevation maps of the Moon. The crater database was built by simulating images for the nominal trajectory and then running them through the CDA to detect craters.

The preliminary results of one of these simulations, shown in Fig. 12, show that the DLT triangulation of the spacecraft position is able to use *a priori* position to shift the distribution of state error closer to zero mean and usually closer to the true simulated position. The right side of Fig. 12 also shows that the detected crater image locations are close to the true locations predicted by the database and that the image locations predicted by the DLT-estimated spacecraft position are closer to the truth than the *a priori*. These position estimate residuals are similar in distribution to the Monte Carlo results of triangulated position residuals presented in Fig. 10, which indicates this is within expected behavior for the CIA. These results indicate the CDA and CIA are working as intended and are able to correctly identify craters in the images.

CONCLUSIONS

Lunar craters offer unique features to utilize for terrain relative navigation (TRN), and are often a convenient choice due to their prevalence on the Moon and the thorough documentation of their positions. These advantages motivated our exploration of a crater-based TRN solution for scenarios when *a priori* state information and optical images of the lunar surface are available. The algorithm provided addresses crater identification through triangulation of the spacecraft state and comparison of projected crater geometry. By strategically testing triads in the crater catalog against triads in the image, the crater identification algorithm (CIA) can efficiently associate multiple pixel coordinates (of crater centers in the image) to 3D points in the world frame. The results indicated that with careful selection of algorithm parameters, false positives can be rejected at the expense of a slight increase in likelihood of no match being found. The results additionally revealed that the CIA was robust to state error. However, inaccuracies in the measured crater centers had a significant effect on

the performance of the CIA, indicating that error caused by the crater detection algorithms (CDA) is important. Work is underway to verify and validate this CIA performance in the context of the IM-2 mission architecture, but initial results show promise for a simple, crater-based TRN scheme for lunar lander navigation.

REFERENCES

- [1] J. A. Christian, L. Hong, P. McKee, R. Christensen, and T. P. Crain, “Image-Based Lunar Terrain Relative Mavigation Without a Map: Measurements,” *Journal of Spacecraft and Rockets*, Vol. 58, 2021, pp. 164–181.
- [2] A. E. Johnson, Y. Cheng, N. Trawny, J. F. Montgomery, S. Schroeder, J. Chang, D. Clouse, S. Aaron, and S. Mohan, “Implementation of a Map Relative Localization System for Planetary Landing,” *Journal of Guidance, Control, and Dynamics*, Vol. 46, No. 4, 2023, pp. 618–637.
- [3] T. Driver, K. Skinner, M. Dor, and P. Tsiotras, “AstroVision: Towards Autonomous Feature Detection and Description for Missions to Small Bodies Using Deep Learning,” *Acta Astronautica*, Vol. 210, 2023, pp. 393–410.
- [4] R. D. Olds, C. J. Miller, C. D. Norman, C. E. Mario, K. Berry, E. Palmer, O. S. Barnouin, M. G. Daly, J. R. Weirich, J. A. Seabrook, C. A. Bennett, D. Lorenz, B. Rizk, B. J. Bos, and D. S. Lauretta, “The Use of Digital Terrain Models for Natural Feature Tracking at Asteroid Bennu,” *The Planetary Science Journal*, Vol. 3, 5 2022.
- [5] C. Hanak, T. Crain, and R. Bishop, “Crater Identification Algorithm for the Lost in Low Lunar Orbit Scenario,” *AAS Guidance & Control Conference*, No. AAS 10-052, 2008.
- [6] J. A. Christian, H. Derksen, and R. Watkins, “Lunar Crater Identification in Digital Images,” *The Journal of the Astronautical Sciences*, Vol. 68, No. 4, 2021, pp. 1056–1144.
- [7] M. Krause, J. Price, and J. A. Christian, “Analytical Methods in Crater Rim Fitting and Pattern Recognition,” *AAS/AIAA Astrodynamics Specialist Conference*, No. AAS 23-362, 2023.
- [8] S. Henry and J. A. Christian, “Absolute Triangulation Algorithms for Space Exploration,” *Journal of Guidance, Control, and Dynamics*, Vol. 46, No. 1, 2023, pp. 21–46.
- [9] J. A. Christian, “A Tutorial on Horizon-Based Optical Navigation and Attitude Determination With Space Imaging Systems,” *IEEEAccess*, Vol. 9, 2021, pp. 19819–19824.
- [10] J. A. Christian and H. Derksen, “Invariant Theory as a Tool for Spacecraft Navigation,” *AAS/AIAA Astrodynamics Specialist Conference, Charlotte, NC*, No. AAS 22-746, 2022.
- [11] D. Arnas, M. A. Fialho, and D. Mortari, “Fast and Robust Kernel Generators for Star Trackers,” *Acta Astronautica*, Vol. 46, 2017, pp. 291–302.
- [12] B. Maass, S. Woicke, W. M. Oliveira, B. Razgus, and H. Krüger, “Crater Navigation System for Autonomous Precision Landing on the Moon,” *Journal of Guidance, Control, and Dynamics*, Vol. 43, No. 8, 2020, pp. 1414–1431.
- [13] S. J. Robbins, “A New Global Database of Lunar Impact Craters $>1\text{--}2\text{ km}$: 1. Crater Locations and Sizes, Comparisons With Published Databases, and Global Analysis,” *Journal of Geophysical Research: Planets*, Vol. 124, 2018, p. 871–892.