ANALYTICAL METHODS IN CRATER RIM FITTING AND PATTERN RECOGNITION

Michael Krause, Jonathan Price, and John Christian[‡]

Upcoming lunar missions are expected to utilize optical measurements for navigation. View invariants enable "lost-in-space" terrain relative navigation about cratered celestial bodies. Lunar crater rims form ellipses when imaged. Noisy measurements of points lying along this rim will be obtained, and may be fit to an ellipse to calculate the invariants. Hyper least squares (HLS) provides an attractive performance benefit compared to traditional (or total) least squares for this task. An analytical derivation of the covariance of these invariants in the presence of noise is presented, and is used to analyze performance of the invariants in realistic situations.

INTRODUCTION

View invariants provide a mathematical framework¹ for recognizing patterns of craters for "lostin-space" terrain relative navigation (TRN) around the Moon² and other cratered celestial bodies. It is well known that lunar craters have predominantly ellipse-shaped rims, due to the underlying physical processes of the impacts that shape their creation.^{3,4} Furthermore, when these craters are imaged by a conventional camera (i.e., not a pushbroom-type camera), these elliptical rims will be projected into ellipses in the resultant image.² For a pattern of two or more nearly coplanar crater rims, their mathematical representations as ellipses can be combined to form a series of invariant values, which are independent of the pose of the imager. Figure 1 provides an illustration of this concept using imagery of the lunar surface from Artemis I.

Utilizing these invariants, an autonomous optical navigation (OPNAV) pipeline for a spacecraft operating about a cratered body may then be envisaged. To begin, the spacecraft would capture multiple images of the surface (notably, without any constraints on pointing direction, except that a desired quantity of craters are visible). Then, points lying on crater rims would be identified via image processing techniques of choice, which likely would include a pixel-level rim detection scheme coupled with a subpixel refinement routine.⁵ Following this, these points would be fit to an ellipse, and these ellipses would be used to generate crater pattern invariants. Finally, assume that the spacecraft is equipped with a database of known invariant values for selected crater patterns on the target body's surface, and their associated locations. By matching the observed crater patterns to those in the database, known features are established across a set of images, and standard pose estimation algorithms may be employed to obtain the state of the spacecraft.² Note that this matching and subsequent pose estimation does not necessitate any *a priori* knowledge of the state of the spacecraft, justifying the classification of such a pipeline as a "lost-in-space" navigation tool.¹

^{*}Graduate Research Assistant, Guggenheim School of Aerospace Engineering, Georgia Institute of Technology, Atlanta, GA, USA.

[†]Undergraduate Research Assistant, Guggenheim School of Aerospace Engineering, Georgia Institute of Technology, Atlanta, GA, USA.

^{*}Associate Professor, Guggenheim School of Aerospace Engineering, Georgia Institute of Technology, Atlanta, GA, USA.



Figure 1 An example of crater pattern invariants calculated using publicly available imagery from Artemis I's OPNAV camera. a) Image art001e002630 with area of interest indicated. b) Image art001e002610 with area of interest indicated. Both areas of interest capture the same three craters, but from different poses. c, d) Cropped and zoomed versions of (a) and (b), respectively, overlaid with calculations of crater pattern invariant values between all pairs of craters. Note that noise ensures these values are not identical across images, and this work investigates the effect of this noise on these invariant values. These images are publicly available through the NASA/JSC Flickr page: https://www.flickr.com/photos/nasa2explore

This work addresses the effect of sensor noise on this pipeline in two aspects: ellipse fitting algorithms and a covariance analysis of crater pattern invariants, illustrated via Figure 2c and 2d, respectively.

When presented with a set of noisy data points that require an ellipse fit, it is common to simply apply a Least-Squares (LS) approach. LS provides a straightforward path for constraining a conic fitting problem to have elliptical solutions, and is relatively simple to implement.⁶ These reasons have likely led to its popularity, however this approach for ellipse fitting is known to produce a biased estimate.⁷ Techniques for producing unbiased ellipse fits have been discussed in mathematical and computer vision communities,^{8–10} but do not seem to have gained a significant foothold amongst OPNAV practitioners—a point that this work aims to resolve. Specifically, Kanatani and Rangarajan's "hyper least squares" method provides an unbiased ellipse fit estimate up to second order noise terms.¹¹ This particular method is especially attractive given that it is non-iterative, unlike other unbiased estimators such as the popular Approximate Maximum Likelihood (AML) methods.¹² A derivation of the HLS algorithm in terms more familiar to the OPNAV community is



Figure 2 A cartoon representation of the envisioned autonomous OPNAV pipeline. a) Crater rims will characteristically form ellipses, and two of these ellipses may be combined to form a pair of crater pattern invariants. b) When imaging these craters, a spacecraft will only have access to noisy measurements of points lying on the rims. c) These noisy measurements are fit to ellipses, which will not generally match the "truth" ellipses of the crater rims. d) Crater pattern invariants are computed from these fitted ellipses, and will not generally match the "truth" pattern invariants. e) These invariant features are matched against a database, enabling pose estimation across multiple observations. This work primarily focuses on steps (c) and (d).

provided in this manuscript. As a component of this, there is discussion concerning the covariance of these ellipse fits in terms of the noise present in the crater rim points.

This analysis is further extended with a discussion of the covariance of crater pattern invariants in the presence of noise. An analytical derivation of this covariance is provided for a pattern of two craters. Monte Carlo simulations are then conducted to numerically verify this covariance formulation. From an application standpoint, this covariance can then be leveraged to produce a statistic to assess the confidence that a crater pattern invariant is indeed a match across two images. These techniques are then applied to real world imagery from Artemis I and the Dawn mission at Ceres, and statistical results are discussed.

DERIVATION OF HYPER LEAST SQUARES ELLIPSE FITTING

Kanatani and Rangarajan¹¹ describe an algebraic conic fitting method that they refer to as "hyper least squares" (HLS), which couples a least squares (LS) approach with a carefully selected normalization term that yields an unbiased estimate up to second order noise terms. We now present an expanded discussion the HLS derivation from Ref. [11] in order to familiarize spacecraft navigators

with its advantages.

The implicit equation for a conic is

$$Ax_i^2 + Bx_iy_i + Cy_i^2 + Dx_i + Fy_i + G = 0$$
(1)

A given conic is an ellipse if $B^2 - 4AC < 0$. If we define

$$\boldsymbol{a}^{T} = \begin{bmatrix} A & B & C & D & F & G \end{bmatrix}$$
(2)

$$\boldsymbol{\xi}_{i}^{T} = \begin{bmatrix} x_{i}^{2} & x_{i}y_{i} & y_{i}^{2} & x_{i} & y_{i} & 1 \end{bmatrix}$$
(3)

then we can compactly represent Eq. (1) as a linear system,

$$\boldsymbol{\xi}_i^T \boldsymbol{a} = 0 \tag{4}$$

Now, suppose we wish to fit an ellipse to a set of n two-dimensional (2-D) points $\{x_i\}_{i=1}^n$ defined as

$$\boldsymbol{x}_i = \begin{bmatrix} x_i \\ y_i \end{bmatrix} \tag{5}$$

The true points $\{x_i\}_{i=1}^n$ are rarely available to us in practice. Instead, suppose that we only have access to noisy observations of these points,

$$\tilde{\boldsymbol{x}}_{i} = \begin{bmatrix} \tilde{x}_{i} \\ \tilde{y}_{i} \end{bmatrix} = \boldsymbol{x}_{i} + \boldsymbol{\nu}_{i} = \begin{bmatrix} x_{i} \\ y_{i} \end{bmatrix} + \begin{bmatrix} \nu_{x_{i}} \\ \nu_{y_{i}} \end{bmatrix}$$
(6)

where $\boldsymbol{\nu}_i \sim \mathcal{N}(0, \boldsymbol{R}_{\boldsymbol{x}_i})$. Assuming that the image processing errors are nearly isotropic, we may approximate the point covariance as $\boldsymbol{R}_{\boldsymbol{x}_i} = E[\boldsymbol{\nu}_i \boldsymbol{\nu}_i^T] \approx \sigma_{\boldsymbol{x}_i}^2 \boldsymbol{I}_{2\times 2}$, where $E[\cdot]$ is the expected value operator.

In turn, let us study how this addition of noise affects our ξ_i vector. We begin by defining its noisy counterpart,

$$\tilde{\boldsymbol{\xi}}_{i} = \begin{bmatrix} \tilde{x}_{i}^{2} \\ \tilde{x}_{i} \tilde{y}_{i} \\ \tilde{y}_{i}^{2} \\ \tilde{x}_{i} \\ \tilde{y}_{i} \\ 1 \end{bmatrix} = \begin{bmatrix} (x_{i} + \nu_{x_{i}})^{2} \\ (x_{i} + \nu_{x_{i}})(y_{i} + \nu_{y_{i}}) \\ (y_{i} + \nu_{y_{i}})^{2} \\ (x_{i} + \nu_{x_{i}}) \\ (y_{i} + \nu_{y_{i}}) \\ 1 \end{bmatrix}$$
(7)

We can then expand this out and group by the order of the small noise terms due to u_i

$$\tilde{\boldsymbol{\xi}}_{i} = \boldsymbol{\xi}_{i} + \delta_{1}\boldsymbol{\xi}_{i} + \delta_{2}\boldsymbol{\xi}_{i} = \begin{bmatrix} x_{i}^{2} \\ x_{i}y_{i} \\ y_{i}^{2} \\ x_{i} \\ y_{i} \\ 1 \end{bmatrix} + \begin{bmatrix} 2x_{i}\nu_{x_{i}} \\ y_{i}\nu_{x_{i}} + x_{i}\nu_{y_{i}} \\ 2y_{i}\nu_{y_{i}} \\ \nu_{x_{i}} \\ \nu_{y_{i}} \\ 0 \end{bmatrix} + \begin{bmatrix} \nu_{x_{i}}^{2} \\ \nu_{x_{i}}\nu_{y_{i}} \\ 0 \\ 0 \\ 0 \end{bmatrix}$$
(8)

At this point, we note that the first order term may be written as

$$\delta_1 \boldsymbol{\xi}_i = \left(\frac{\partial \boldsymbol{\xi}_i}{\partial \boldsymbol{x}_i}\right) \boldsymbol{\nu}_i \tag{9}$$

where we've analytically computed the partial derivative as

$$\frac{\partial \boldsymbol{\xi}_{i}}{\partial \boldsymbol{x}_{i}} = \begin{bmatrix} 2x_{i} & 0\\ y_{i} & x_{i}\\ 0 & 2y_{i}\\ 1 & 0\\ 0 & 1\\ 0 & 0 \end{bmatrix}$$
(10)

Utilizing this fact, we can then find the covariance of ξ_i , which is computed as

$$\boldsymbol{R}_{\boldsymbol{\xi}_{i}} = E[\delta_{1}\boldsymbol{\xi}_{i}\,\delta_{1}\boldsymbol{\xi}_{i}^{T}] \approx \left(\frac{\partial\boldsymbol{\xi}_{i}}{\partial\boldsymbol{x}_{i}}\right) E[\boldsymbol{\nu}_{i}\boldsymbol{\nu}_{i}^{T}] \left(\frac{\partial\boldsymbol{\xi}_{i}}{\partial\boldsymbol{x}_{i}}\right)^{T} = \sigma_{\boldsymbol{x}_{i}}^{2} \left(\frac{\partial\boldsymbol{\xi}_{i}}{\partial\boldsymbol{x}_{i}}\right) \left(\frac{\partial\boldsymbol{\xi}_{i}}{\partial\boldsymbol{x}_{i}}\right)^{T}$$
(11)

One may then compute

$$\boldsymbol{R}_{\boldsymbol{\xi}_{i}} \approx \sigma_{\boldsymbol{x}_{i}}^{2} \left(\frac{\partial \boldsymbol{\xi}_{i}}{\partial \boldsymbol{x}_{i}}\right) \left(\frac{\partial \boldsymbol{\xi}_{i}}{\partial \boldsymbol{x}_{i}}\right)^{T} = \sigma_{\boldsymbol{x}_{i}}^{2} \boldsymbol{R}_{0,\boldsymbol{\xi}_{i}} = \sigma_{\boldsymbol{x}_{i}}^{2} \begin{bmatrix} 4x_{i}^{2} & 2x_{i}y_{i} & 0 & 2x_{i} & 0 & 0\\ 2x_{i}y_{i} & x_{i}^{2} + y_{i}^{2} & 2x_{i}y_{i} & y_{i} & x_{i} & 0\\ 0 & 2x_{i}y_{i} & 4y_{i}^{2} & 0 & 2y_{i} & 0\\ 2x_{i} & y_{i} & 0 & 1 & 0 & 0\\ 0 & x_{i} & 2y_{i} & 0 & 1 & 0\\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$
(12)

which, as we expect, is rank deficient (6×6 matrix of rank 5).

Returning our attention to the task of ellipse fitting, we note that when noisy measurements are used in Eq. (4), we do not obtain an equality with zero. Instead,

$$\tilde{\boldsymbol{\xi}}_{i}^{T}\boldsymbol{a} = \boldsymbol{\epsilon}_{i} \neq 0 \tag{13}$$

It follows that we should solve for *a* from many noisy observations by minimizing the sum of the squares of ϵ_i . This is captured mathematically by the cost function

$$\min J(\boldsymbol{a}) = \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{\epsilon}_{i}^{T} \boldsymbol{\epsilon}_{i} = \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{a}^{T} \tilde{\boldsymbol{\xi}}_{i} \tilde{\boldsymbol{\xi}}_{i}^{T} \boldsymbol{a}$$
(14)

By defining the matrices M and \tilde{M} as

$$\boldsymbol{M} = \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{\xi}_{i} \boldsymbol{\xi}_{i}^{T} \quad \text{and} \quad \tilde{\boldsymbol{M}} = \frac{1}{n} \sum_{i=1}^{n} \tilde{\boldsymbol{\xi}}_{i} \tilde{\boldsymbol{\xi}}_{i}^{T}$$
(15)

the cost function may be written as

$$\min J(\boldsymbol{a}) = \boldsymbol{a}^T \tilde{\boldsymbol{M}} \boldsymbol{a} \tag{16}$$

Now, we observe that the scale of the vector \boldsymbol{a} is arbitrary and \boldsymbol{a} describes the same conic as $t\boldsymbol{a}$ for $t \in \mathbb{R}_{\neq 0}$. It is well-known that different choices for constraining the length of \boldsymbol{a} lead to different error statistics—with some being more or less biased than others.⁷ Suppose that we describe this normalization with a quadratic constraint of the form

$$\boldsymbol{a}^T \boldsymbol{N} \boldsymbol{a} = c \tag{17}$$

where N is assumed to be symmetric. This constraint may be adjoined to our cost function with a Lagrange multiplier to arrive at

min
$$J(\boldsymbol{a}) = \boldsymbol{a}^T \tilde{\boldsymbol{M}} \boldsymbol{a} + \lambda (c - \boldsymbol{a}^T \boldsymbol{N} \boldsymbol{a})$$
 (18)

Applying the first differential condition, and noting that *M* and *N* are both symmetric,

$$2\dot{M}a - 2\lambda Na = 0 \tag{19}$$

which leads to a generalized eigenvalue problem of the form

$$Ma = \lambda Na \tag{20}$$

where our desired solution a will be the generalized eigenvector with the smallest magnitude.

Recall that our desired goal is to obtain an unbiased estimate of a. Selection of an appropriate normalization N will yield such an estimate. Note that if N is simply chosen to be the identity matrix, I, this results in a standard LS solution. Taubin's method, another well-known normalization scheme for ellipse fitting that tends to outperform LS (but is still statistically biased),¹¹ chooses N to be:

$$N_T = \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{R}_{0,\boldsymbol{\xi}_i} \tag{21}$$

Since the matrix \dot{M} is noisy, the generalized eigenvectors produced by Eq. (20) are also noisy. Thus, we proceed by conducting an analysis of the effect of noise on the matrix M (and thus, on the resultant generalized eigenvector a) to help inform this choice of normalization and achieve an unbiased estimate.

We begin by looking at the perturbations in *M* to second order in ν_i ,

$$\tilde{\boldsymbol{M}} = \sum_{i=1}^{n} \tilde{\boldsymbol{\xi}}_{i} \tilde{\boldsymbol{\xi}}_{i}^{T} = \sum_{i=1}^{n} (\boldsymbol{\xi}_{i} + \delta_{1} \boldsymbol{\xi}_{i} + \delta_{2} \boldsymbol{\xi}_{i}) (\boldsymbol{\xi}_{i} + \delta_{1} \boldsymbol{\xi}_{i} + \delta_{2} \boldsymbol{\xi}_{i})^{T} = \boldsymbol{M} + \delta_{1} \boldsymbol{M} + \delta_{2} \boldsymbol{M} + \mathcal{O}(\|\boldsymbol{\nu}_{i}\|^{3})$$
(22)

where one may explicitly compute the first and second order perturbations as

$$\delta_1 \boldsymbol{M} = \frac{1}{n} \sum_{i=1}^n \left(\boldsymbol{\xi}_i \delta_1 \boldsymbol{\xi}_i^T + \delta_1 \boldsymbol{\xi}_i \boldsymbol{\xi}_i^T \right)$$
(23)

$$\delta_2 \boldsymbol{M} = \frac{1}{n} \sum_{i=1}^n \left(\boldsymbol{\xi}_i \delta_2 \boldsymbol{\xi}_i^T + \delta_1 \boldsymbol{\xi}_i \delta_1 \boldsymbol{\xi}_i^T + \delta_2 \boldsymbol{\xi}_i \boldsymbol{\xi}_i^T \right)$$
(24)

Furthermore, we expand terms in Eq. (20) to second order:

$$(\boldsymbol{M} + \delta_1 \boldsymbol{M} + \delta_2 \boldsymbol{M} + \ldots)(\boldsymbol{a} + \delta_1 \boldsymbol{a} + \delta_2 \boldsymbol{a} + \ldots) = (\lambda + \delta_1 \lambda + \delta_2 \lambda + \ldots)N(\boldsymbol{a} + \delta_1 \boldsymbol{a} + \delta_2 \boldsymbol{a} + \ldots)$$
(25)

Note that N is not expanded since it is a normalization factor to be determined. From this expansion, we can group terms by their order, which results in the following relationships:

$$Ma = \lambda Na \tag{26}$$

$$\boldsymbol{M}\delta_{1}\boldsymbol{a} + \delta_{1}\boldsymbol{M}\boldsymbol{a} = \lambda \boldsymbol{N}\delta_{1}\boldsymbol{a} + \delta_{1}\lambda \boldsymbol{N}\boldsymbol{a}$$
(27)

$$\boldsymbol{M}\delta_{2}\boldsymbol{a} + \delta_{1}\boldsymbol{M}\delta_{1}\boldsymbol{a} + \delta_{2}\boldsymbol{M}\boldsymbol{a} = \lambda \boldsymbol{N}\delta_{2}\boldsymbol{a} + \delta_{1}\lambda \boldsymbol{N}\delta_{1}\boldsymbol{a} + \delta_{2}\lambda \boldsymbol{N}\boldsymbol{a}$$
(28)

Combining Eqs. (4) and (15), we see that in a noiseless case, Ma = 0. This directly implies from Eq. (26) that $\lambda = 0$. We yet again employ Eq. (4) and combine it with Eq. (23), finding that $a^T \delta_1 Ma = 0$ as well. Then, multiplying Eq. (27) by a^T from the left and applying these conditions:

$$0 = \boldsymbol{a}^T \delta_1 \lambda \boldsymbol{N} \boldsymbol{a} \to \delta_1 \lambda = 0 \tag{29}$$

Utilizing Eq. (27) once more and keeping in mind this new condition, when the pseudoinverse M^+ is multiplied from the left, then we obtain

$$\delta_1 \boldsymbol{a} = -\boldsymbol{M}^+ \delta_1 \boldsymbol{M} \boldsymbol{a} \tag{30}$$

Plugging Eq. (30) into Eq. (28) and rearranging then yields

$$\delta_2 \lambda = \frac{a^T \delta_2 M a - a^T \delta_1 M M^+ \delta_1 M a}{a^T N a}$$
(31)

or, letting

$$\boldsymbol{T} = \delta_2 \boldsymbol{M} - \delta_1 \boldsymbol{M} \boldsymbol{M}^+ \delta_1 \boldsymbol{M} \tag{32}$$

then,

$$\delta_2 \lambda = \frac{\boldsymbol{a}^T \boldsymbol{T} \boldsymbol{a}}{\boldsymbol{a}^T \boldsymbol{N} \boldsymbol{a}} \tag{33}$$

The next term of interest is $\delta_2 a$, the second order error in the ellipse fit. It is important to note at this point that the length of a will be constrained by Eq. (17). However, in order to find an unbiased estimate to the generalized eigenvalue problem, we will further scale a to have unit norm. Thus, since a is always a unit vector and will not have an error in magnitude as a result, we seek to find the second order error orthogonal to a, i.e., $\delta_2^{\perp} a$. Further, we note that a is, by definition, the null vector of M, and thus M^+M is an orthogonal projection along a. Therefore,

$$\delta_2^{\perp} \boldsymbol{a} = (\boldsymbol{M}^+ \boldsymbol{M}) \delta_2 \boldsymbol{a} \tag{34}$$

Now, we multiply Eq. (28) from the left by M^+ and substitute in Eq. (30). After rearranging, this yields

$$\delta_2^{\perp} \boldsymbol{a} = \delta_2 \lambda \boldsymbol{M}^+ \boldsymbol{N} \boldsymbol{a} + \boldsymbol{M}^+ \delta_1 \boldsymbol{M} \boldsymbol{M}^+ \delta_1 \boldsymbol{M} \boldsymbol{a} - \boldsymbol{M}^+ \delta_2 \boldsymbol{M} \boldsymbol{a} = \frac{\boldsymbol{a}^T \boldsymbol{T} \boldsymbol{a}}{\boldsymbol{a}^T \boldsymbol{N} \boldsymbol{a}} \boldsymbol{M}^+ \boldsymbol{N} \boldsymbol{a} - \boldsymbol{M}^+ \boldsymbol{T} \boldsymbol{a}$$
(35)

Ultimately, we seek an expression for N such that $\delta_2^{\perp} a = 0$. Therefore, we compute the expectation of $\delta_2^{\perp} a$ with the goal of manipulating that accordingly,

$$E[\delta_2^{\perp} \boldsymbol{a}] = \boldsymbol{M}^+ \left[\frac{\boldsymbol{a}^T E[\boldsymbol{T}] \boldsymbol{a}}{\boldsymbol{a}^T N \boldsymbol{a}} N \boldsymbol{a} - E[\boldsymbol{T}] \boldsymbol{a} \right]$$
(36)

It is now elementary to see that if N = E[T], the result is that $E[\delta_2^{\perp} a] = 0$.

Computing the expectation of T may be done term-by-term. First, invoking Eq. (24),

$$E[\delta_2 \boldsymbol{M}] = \frac{1}{n} \sum_{i=1}^n \left(\boldsymbol{\xi}_i E[\delta_2 \boldsymbol{\xi}_i]^T + E[\delta_1 \boldsymbol{\xi}_i \delta_1 \boldsymbol{\xi}_i^T] + E[\delta_2 \boldsymbol{\xi}_i] \boldsymbol{\xi}_i^T \right)$$
(37)

where (recalling Eq. (8))

$$E[\delta_2 \boldsymbol{\xi}_i] = \sigma_{\boldsymbol{x}_i}^2 \boldsymbol{e} \tag{38}$$

where

$$\boldsymbol{e} = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}^T$$
(39)

As a final prerequisite, let

$$\boldsymbol{\xi}_c = \frac{1}{n} \sum_{i=1}^n \boldsymbol{\xi}_i \tag{40}$$

Then, combining with Eqs. (12) and (21), assuming that the noise will be characterized identically for all sampled points, and introducing the symmetrization operator $S[B] = (B + B^T)/2$, Eq. (37) may be compactly written as

$$E[\delta_2 \boldsymbol{M}] = \sigma^2 (\boldsymbol{N}_T + 2\mathcal{S}[\boldsymbol{\xi}_c \boldsymbol{e}])$$
(41)

Next, the expectation of the second term of T may be found as:

$$E\left[\delta_{1}\boldsymbol{M}\boldsymbol{M}^{+}\delta_{1}\boldsymbol{M}\right] = \frac{\sigma^{2}}{n^{2}}\sum_{i=1}^{n}\left(\operatorname{tr}\left[\boldsymbol{M}^{+}\boldsymbol{R}_{0,\boldsymbol{\xi}_{i}}\right]\boldsymbol{\xi}_{i}\boldsymbol{\xi}_{i}^{T} + \boldsymbol{\xi}_{i}^{T}\boldsymbol{M}^{+}\boldsymbol{\xi}_{i}\boldsymbol{R}_{0,\boldsymbol{\xi}_{i}} + 2\mathcal{S}\left[\boldsymbol{R}_{0,\boldsymbol{\xi}_{i}}\boldsymbol{M}^{+}\boldsymbol{\xi}_{i}\boldsymbol{\xi}_{i}^{T}\right]\right) \quad (42)$$

The interested reader is directed to Appendix B of Kanatani and Rangarajan's derivation¹¹ for complete details concerning this derivation.

Taken collectively, then:

$$E[\mathbf{T}] = E[\delta_2 \mathbf{M}] - E\left|\delta_1 \mathbf{M} \mathbf{M}^+ \delta_1 \mathbf{M}\right|$$
(43)

and noting that the σ^2 term is allowed to drop out without loss of generality, this leads to the selection of N as

$$\boldsymbol{N} = \boldsymbol{N}_{T} + 2\boldsymbol{\mathcal{S}}[\boldsymbol{\xi}_{c}\boldsymbol{e}] - \frac{1}{n^{2}} \sum_{i=1}^{n} \left(\operatorname{tr} \left[\boldsymbol{M}^{+}\boldsymbol{R}_{0,\boldsymbol{\xi}_{i}} \right] \boldsymbol{\xi}_{i}\boldsymbol{\xi}_{i}^{T} + \boldsymbol{\xi}_{i}^{T}\boldsymbol{M}^{+}\boldsymbol{\xi}_{i}\boldsymbol{R}_{0,\boldsymbol{\xi}_{i}} + 2\boldsymbol{\mathcal{S}} \left[\boldsymbol{R}_{0,\boldsymbol{\xi}_{i}}\boldsymbol{M}^{+}\boldsymbol{\xi}_{i}\boldsymbol{\xi}_{i}^{T} \right] \right) \quad (44)$$

The magnitude of the final term is largely insignificant compared to the first two when the number of sampled points, n, is high. Kanatani and Rangarajan propose that this final term may be dropped in this regime, yielding an algoritm they term "Semi-hyper Least Squares", or SHLS. We continue the derivation of the full form of HLS, but note that SHLS provides a result that is oftentimes indistinguishable when many sample points are available, and requires less computational effort.

When only noisy points are available, then these must be used to calculate N. Specifically, ξ replaces ξ , $\tilde{\xi}_c$ replaces ξ_c , \tilde{M} replaces M, and $R_{0,\tilde{\xi}_i}$ replaces R_{0,ξ_i} . Note that, in general, this makes \tilde{M} nonsingular, though we desire it to be singular (recall that a lies in the null space of M). Thus, prior to using \tilde{M} to compute N, we replace the smallest eigenvalue of \tilde{M} with zero by spectral decomposition.

We see that the choice of N affects the bias, but not the standard deviation. In particular, we may compute the covariance of a as $P_a = E[\delta_1 a \delta_1 a^T]$. By substitution from Eq. (30) this is

$$\boldsymbol{P}_{\boldsymbol{a}} = \boldsymbol{M}^{+} \boldsymbol{E} \left[\delta_{1} \boldsymbol{M} \boldsymbol{a} \boldsymbol{a}^{T} \delta_{1} \boldsymbol{M}^{T} \right] \boldsymbol{M}^{+T}$$
(45a)

$$= \mathbf{M}^{+} E \left[\left(\sum_{i=1}^{n} \boldsymbol{\xi}_{i} \delta_{1} \boldsymbol{\xi}_{i}^{T} + \delta_{1} \boldsymbol{\xi}_{i} \boldsymbol{\xi}_{i}^{T} \right) \mathbf{a} \mathbf{a}^{T} \left(\sum_{j=1}^{n} \boldsymbol{\xi}_{j} \delta_{1} \boldsymbol{\xi}_{j}^{T} + \delta_{1} \boldsymbol{\xi}_{j} \boldsymbol{\xi}_{j}^{T} \right)^{T} \right] \mathbf{M}^{+T}$$
(45b)

$$= \mathbf{M}^{+} E \left[\left(\sum_{i=1}^{n} \boldsymbol{\xi}_{i} \delta_{1} \boldsymbol{\xi}_{i}^{T} \mathbf{a} \right) \left(\sum_{j=1}^{n} \boldsymbol{\xi}_{j} \delta_{1} \boldsymbol{\xi}_{j}^{T} \mathbf{a} \right)^{T} \right] \mathbf{M}^{+T}$$
(45c)

$$= \mathbf{M}^{+} E \left[\left(\sum_{i=1}^{n} \boldsymbol{\xi}_{i} \boldsymbol{a}^{T} \delta_{1} \boldsymbol{\xi}_{i} \right) \left(\sum_{j=1}^{n} \boldsymbol{\xi}_{j} \boldsymbol{a}^{T} \delta_{1} \boldsymbol{\xi}_{j} \right)^{T} \right] \mathbf{M}^{+T}$$
(45d)

$$= \boldsymbol{M}^{+} \left(\sum_{i=1}^{n} \sum_{j=1}^{n} \boldsymbol{\xi}_{i} \boldsymbol{a}^{T} E\left[\delta_{1} \boldsymbol{\xi}_{i} \delta_{1} \boldsymbol{\xi}_{j}^{T} \right] \boldsymbol{a} \boldsymbol{\xi}_{j}^{T} \right) \boldsymbol{M}^{+T}$$
(45e)

$$= \boldsymbol{M}^{+} \left(\sum_{i=1}^{n} \boldsymbol{\xi}_{i} \boldsymbol{a}^{T} \boldsymbol{R}_{\boldsymbol{\xi}_{i}} \boldsymbol{a} \boldsymbol{\xi}_{i}^{T} \right) \boldsymbol{M}^{+T}$$
(45f)

$$= \boldsymbol{M}^{+} \left(\sum_{i=1}^{n} (\boldsymbol{a}^{T} \boldsymbol{R}_{\boldsymbol{\xi}_{i}} \boldsymbol{a}) (\boldsymbol{\xi}_{i} \boldsymbol{\xi}_{i}^{T}) \right) \boldsymbol{M}^{+T}$$
(45g)

$$= \boldsymbol{M}^{+} \left(\sum_{i=1}^{n} (\boldsymbol{a}^{T} \boldsymbol{R}_{\boldsymbol{\xi}_{i}} \boldsymbol{a})(\boldsymbol{\xi}_{i} \boldsymbol{\xi}_{i}^{T}) \right) \boldsymbol{M}^{+}$$
(45h)

SENSITIVITY OF CRATER PATTERN INVARIANTS

Obtaining an analytical covariance for crater pattern invariants is a critical step for enabling useful application of this OPNAV framework with real world imagery. Specifically, this work focuses on the covariance for a pair of crater pattern invariants. Once obtained, this covariance may then be used to form a statistic to rigorously compare crater patterns across images. Knowledge of such a covariance depends on the covariance of the ellipse fits used for generating the invariants – as demonstrated, itself a function of the ellipse parameters and the noise inherent in the imaging and image processing system.

Until this point, the ellipse fit coefficients have been treated as a 6 element vector, as defined in Eq. (2). For the computations of the invariants (and thus, for the computation of their covariance), it is necessary to place this into a symmetric matrix form instead, defined as

$$\mathbf{A} = \begin{bmatrix} A & B/2 & D/2 \\ B/2 & C & F/2 \\ D/2 & F/2 & G \end{bmatrix}$$
(46)

The invariants for a pair of nearly co-planar craters are²

$$I_{ij} = \left(\frac{\det[\mathbf{A}_i]}{\det[\mathbf{A}_j]}\right)^{1/3} \operatorname{Tr}\left[\mathbf{A}_i^{-1}\mathbf{A}_j\right]$$
(47)

$$I_{ji} = \left(\frac{\det[\boldsymbol{A}_j]}{\det[\boldsymbol{A}_i]}\right)^{1/3} \operatorname{Tr}\left[\boldsymbol{A}_j^{-1}\boldsymbol{A}_i\right]$$
(48)

Considering now the perturbations that this pair of invariants will experience in the presence of noise, we construct

$$\delta \mathbf{y}_{ij} = \begin{bmatrix} \delta I_{ij} \\ \delta I_{ji} \end{bmatrix} = \begin{bmatrix} \partial I_{ij} / \partial \mathbf{a}_i & \partial I_{ij} / \partial \mathbf{a}_j \\ \partial I_{ji} / \partial \mathbf{a}_i & \partial I_{ji} / \partial \mathbf{a}_j \end{bmatrix} \begin{bmatrix} \delta \mathbf{a}_i \\ \delta \mathbf{a}_j \end{bmatrix}$$
(49)

where the covariance is, by definition,

$$\boldsymbol{P}_{\boldsymbol{y}_{ij}} = E[\delta \boldsymbol{y}_{ij} \, \delta \boldsymbol{y}_{ij}^T] \tag{50}$$

To this end, we now seek closed-form expressions for the partial derivatives of each invariant with respect to the ellipse fits, as introduced in Eq. (49).

Begin with the identities:¹³

$$\frac{\partial}{\partial \mathbf{X}} \operatorname{Tr}[\mathbf{A}\mathbf{X}^T] = \mathbf{A}$$
(51)

$$\frac{\partial}{\partial X} \operatorname{Tr}[AX^{-1}B] = -X^{-T}A^{T}B^{T}X^{-T}$$
(52)

$$\frac{\partial}{\partial X} \det[X] = \det[X] X^{-T}$$
(53)

and that $A_i = A_i^T$ and $A_j = A_j^T$, we obtain

$$\frac{\partial I_{ij}}{\partial \mathbf{A}_i} = \operatorname{Tr}[\mathbf{A}_i^{-1}\mathbf{A}_j] \frac{\partial}{\partial \mathbf{A}_i} \left(\frac{\operatorname{det}[\mathbf{A}_i]}{\operatorname{det}[\mathbf{A}_j]}\right)^{1/3} + \left(\frac{\operatorname{det}[\mathbf{A}_i]}{\operatorname{det}[\mathbf{A}_j]}\right)^{1/3} \frac{\partial}{\partial \mathbf{A}_i} \operatorname{Tr}[\mathbf{A}_i^{-1}\mathbf{A}_j]$$
(54a)

$$= \frac{1}{3} \left(\frac{\det[A_i]}{\det[A_j]} \right)^{1/3} \operatorname{Tr}[A_i^{-1}A_j] A_i^{-T} - \left(\frac{\det[A_i]}{\det[A_j]} \right)^{1/3} A_i^{-T} A_j^T A_i^{-T}$$
(54b)

$$= \frac{1}{3} \left(\frac{\det[A_i]}{\det[A_j]} \right)^{1/3} \operatorname{Tr}[A_i^{-1}A_j] A_i^{-1} - \left(\frac{\det[A_i]}{\det[A_j]} \right)^{1/3} A_i^{-1}A_j A_i^{-1}$$
(54c)

$$= \left(\frac{\det[\mathbf{A}_i]}{\det[\mathbf{A}_j]}\right)^{1/3} \mathbf{A}_i^{-1} \left(\frac{1}{3} \operatorname{Tr}[\mathbf{A}_i^{-1}\mathbf{A}_j] \mathbf{I}_{3\times 3} - \mathbf{A}_j \mathbf{A}_i^{-1}\right)$$
(54d)

Furthermore,

$$\frac{\partial I_{ij}}{\partial A_j} = \operatorname{Tr}[A_i^{-1}A_j] \frac{\partial}{\partial A_j} \left(\frac{\operatorname{det}[A_i]}{\operatorname{det}[A_j]}\right)^{1/3} + \left(\frac{\operatorname{det}[A_i]}{\operatorname{det}[A_j]}\right)^{1/3} \frac{\partial}{\partial A_j} \operatorname{Tr}[A_i^{-1}A_j]$$
(55a)

$$= -\frac{1}{3} \left(\frac{\det[\boldsymbol{A}_i]}{\det[\boldsymbol{A}_j]} \right)^{1/3} \operatorname{Tr}[\boldsymbol{A}_i^{-1}\boldsymbol{A}_j] \boldsymbol{A}_j^{-T} + \left(\frac{\det[\boldsymbol{A}_i]}{\det[\boldsymbol{A}_j]} \right)^{1/3} \boldsymbol{A}_i^{-1}$$
(55b)

$$= -\frac{1}{3} \left(\frac{\operatorname{det}[\mathbf{A}_i]}{\operatorname{det}[\mathbf{A}_j]} \right)^{1/3} \operatorname{Tr}[\mathbf{A}_i^{-1}\mathbf{A}_j] \mathbf{A}_j^{-1} + \left(\frac{\operatorname{det}[\mathbf{A}_i]}{\operatorname{det}[\mathbf{A}_j]} \right)^{1/3} \mathbf{A}_i^{-1}$$
(55c)

$$= \left(\frac{\det[A_i]}{\det[A_j]}\right)^{1/3} \left(-\frac{1}{3}\operatorname{Tr}[A_i^{-1}A_j]A_j^{-1} + A_i^{-1}\right)$$
(55d)

Following similar processes for I_{ji} , we obtain,

$$\frac{\partial I_{ji}}{\partial \boldsymbol{A}_i} = \left(\frac{\det[\boldsymbol{A}_j]}{\det[\boldsymbol{A}_i]}\right)^{1/3} \left(-\frac{1}{3}\operatorname{Tr}[\boldsymbol{A}_j^{-1}\boldsymbol{A}_i]\boldsymbol{A}_i^{-1} + \boldsymbol{A}_j^{-1}\right)$$
(56)

and

$$\frac{\partial I_{ji}}{\partial \mathbf{A}_j} = \left(\frac{\det[\mathbf{A}_j]}{\det[\mathbf{A}_i]}\right)^{1/3} \mathbf{A}_j^{-1} \left(\frac{1}{3} \operatorname{Tr}[\mathbf{A}_j^{-1}\mathbf{A}_i] \mathbf{I}_{3\times 3} - \mathbf{A}_i \mathbf{A}_j^{-1}\right)$$
(57)

Recall from Eq. (49) that we seek partial derivatives with respect to the vector of conic coefficients a, rather that with respect to the matrix of coefficients A. We move towards this by applying the vector operator,

$$\frac{\partial I_{ij}}{\partial \text{vec}\left[\boldsymbol{A}_{i}\right]} = \text{vec}\left[\left(\frac{\text{det}[\boldsymbol{A}_{i}]}{\text{det}[\boldsymbol{A}_{j}]}\right)^{1/3} \boldsymbol{A}_{i}^{-1}\left(\frac{1}{3}\operatorname{Tr}[\boldsymbol{A}_{i}^{-1}\boldsymbol{A}_{j}]\boldsymbol{I}_{3\times3} - \boldsymbol{A}_{j}\boldsymbol{A}_{i}^{-1}\right)\right]^{T}$$
(58)

$$\frac{\partial I_{ij}}{\partial \text{vec}\left[\boldsymbol{A}_{j}\right]} = \text{vec}\left[\left(\frac{\text{det}[\boldsymbol{A}_{i}]}{\text{det}[\boldsymbol{A}_{j}]}\right)^{1/3} \left(-\frac{1}{3}\operatorname{Tr}[\boldsymbol{A}_{i}^{-1}\boldsymbol{A}_{j}]\boldsymbol{A}_{j}^{-1} + \boldsymbol{A}_{i}^{-1}\right)\right]^{T}$$
(59)

$$\frac{\partial I_{ji}}{\partial \text{vec}\left[\boldsymbol{A}_{i}\right]} = \text{vec}\left[\left(\frac{\text{det}[\boldsymbol{A}_{j}]}{\text{det}[\boldsymbol{A}_{i}]}\right)^{1/3} \left(-\frac{1}{3}\operatorname{Tr}[\boldsymbol{A}_{j}^{-1}\boldsymbol{A}_{i}]\boldsymbol{A}_{i}^{-1} + \boldsymbol{A}_{j}^{-1}\right)\right]^{T}$$
(60)

$$\frac{\partial I_{ji}}{\partial \text{vec}\left[\boldsymbol{A}_{j}\right]} = \text{vec}\left[\left(\frac{\text{det}[\boldsymbol{A}_{j}]}{\text{det}[\boldsymbol{A}_{i}]}\right)^{1/3} \boldsymbol{A}_{j}^{-1}\left(\frac{1}{3}\operatorname{Tr}[\boldsymbol{A}_{j}^{-1}\boldsymbol{A}_{i}]\boldsymbol{I}_{3\times3} - \boldsymbol{A}_{i}\boldsymbol{A}_{j}^{-1}\right)\right]^{T}$$
(61)

and recognizing that because $vec[A_i] = \Pi a_i$, we obtain

$$\frac{\partial \text{vec}[\mathbf{A}_i]}{\partial \mathbf{a}_i} = \mathbf{\Pi} \tag{62}$$

Using the conventions presented in this work, one may compute Π to be

$$\boldsymbol{\Pi} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1/2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/2 & 0 & 0 \\ 0 & 1/2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1/2 & 0 \\ 0 & 0 & 0 & 0 & 1/2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$
(63)

Now, directly applying the chain rule, we find our partial derivatives of interest

$$\frac{\partial I_{ij}}{\partial \boldsymbol{a}_i} = \operatorname{vec}\left[\left(\frac{\operatorname{det}[\boldsymbol{A}_i]}{\operatorname{det}[\boldsymbol{A}_j]}\right)^{1/3} \boldsymbol{A}_i^{-1} \left(\frac{1}{3}\operatorname{Tr}[\boldsymbol{A}_i^{-1}\boldsymbol{A}_j]\boldsymbol{I}_{3\times 3} - \boldsymbol{A}_j\boldsymbol{A}_i^{-1}\right)\right]^T \boldsymbol{\Pi}$$
(64)

$$\frac{\partial I_{ij}}{\partial \boldsymbol{a}_j} = \operatorname{vec}\left[\left(\frac{\operatorname{det}[\boldsymbol{A}_i]}{\operatorname{det}[\boldsymbol{A}_j]}\right)^{1/3} \left(-\frac{1}{3}\operatorname{Tr}[\boldsymbol{A}_i^{-1}\boldsymbol{A}_j]\boldsymbol{A}_j^{-1} + \boldsymbol{A}_i^{-1}\right)\right]^T \boldsymbol{\Pi}$$
(65)

$$\frac{\partial I_{ji}}{\partial \boldsymbol{a}_i} = \operatorname{vec}\left[\left(\frac{\operatorname{det}[\boldsymbol{A}_j]}{\operatorname{det}[\boldsymbol{A}_i]}\right)^{1/3} \left(-\frac{1}{3}\operatorname{Tr}[\boldsymbol{A}_j^{-1}\boldsymbol{A}_i]\boldsymbol{A}_i^{-1} + \boldsymbol{A}_j^{-1}\right)\right]^T \boldsymbol{\Pi}$$
(66)

$$\frac{\partial I_{ji}}{\partial \boldsymbol{a}_j} = \operatorname{vec}\left[\left(\frac{\operatorname{det}[\boldsymbol{A}_j]}{\operatorname{det}[\boldsymbol{A}_i]}\right)^{1/3} \boldsymbol{A}_j^{-1} \left(\frac{1}{3}\operatorname{Tr}[\boldsymbol{A}_j^{-1}\boldsymbol{A}_i]\boldsymbol{I}_{3\times 3} - \boldsymbol{A}_i\boldsymbol{A}_j^{-1}\right)\right]^T \boldsymbol{\Pi}$$
(67)

Having found these expressions, we can return our attention to the Eq. (50), and compute the expected values,

$$\boldsymbol{P}_{\boldsymbol{y}_{ij}} = E[\delta \boldsymbol{y}_{ij} \, \delta \boldsymbol{y}_{ij}^{T}] = \begin{bmatrix} \partial I_{ij} / \partial \boldsymbol{a}_{i} & \partial I_{ij} / \partial \boldsymbol{a}_{j} \\ \partial I_{ji} / \partial \boldsymbol{a}_{i} & \partial I_{ji} / \partial \boldsymbol{a}_{j} \end{bmatrix} \begin{bmatrix} \boldsymbol{P}_{\boldsymbol{a}_{i}} & \boldsymbol{0}_{6\times 6} \\ \boldsymbol{0}_{6\times 6} & \boldsymbol{P}_{\boldsymbol{a}_{j}} \end{bmatrix} \begin{bmatrix} \partial I_{ij} / \partial \boldsymbol{a}_{i} & \partial I_{ij} / \partial \boldsymbol{a}_{j} \\ \partial I_{ji} / \partial \boldsymbol{a}_{i} & \partial I_{ji} / \partial \boldsymbol{a}_{j} \end{bmatrix}^{T}$$
(68a)

$$= \begin{bmatrix} \sigma_{I_{ij}}^2 & \rho_{ij}\sigma_{I_{ij}}\sigma_{I_{ji}}\\ \rho_{ij}\sigma_{I_{ji}}\sigma_{I_{ij}} & \sigma_{I_{ji}}^2 \end{bmatrix}$$
(68b)

where the four scalar entries in $P_{y_{ij}}$ are

$$\sigma_{I_{ij}}^2 = (\partial I_{ij}/\partial \boldsymbol{a}_i)\boldsymbol{P}_{\boldsymbol{a}_i}(\partial I_{ij}/\partial \boldsymbol{a}_i)^T + (\partial I_{ij}/\partial \boldsymbol{a}_j)\boldsymbol{P}_{\boldsymbol{a}_j}(\partial I_{ij}/\partial \boldsymbol{a}_j)^T$$
(69)

$$\rho_{ij}\sigma_{I_{ji}}\sigma_{I_{ij}} = (\partial I_{ji}/\partial \boldsymbol{a}_i)\boldsymbol{P}_{\boldsymbol{a}_i}(\partial I_{ij}/\partial \boldsymbol{a}_i)^T + (\partial I_{ji}/\partial \boldsymbol{a}_j)\boldsymbol{P}_{\boldsymbol{a}_j}(\partial I_{ij}/\partial \boldsymbol{a}_j)^T$$
(70)

$$\rho_{ij}\sigma_{I_{ij}}\sigma_{I_{ji}} = (\partial I_{ij}/\partial \boldsymbol{a}_i)\boldsymbol{P}_{\boldsymbol{a}_i}(\partial I_{ji}/\partial \boldsymbol{a}_i)^T + (\partial I_{ij}/\partial \boldsymbol{a}_j)\boldsymbol{P}_{\boldsymbol{a}_j}(\partial I_{ji}/\partial \boldsymbol{a}_j)^T$$
(71)

$$\sigma_{I_{ji}}^2 = (\partial I_{ji}/\partial \boldsymbol{a}_i)\boldsymbol{P}_{\boldsymbol{a}_i}(\partial I_{ji}/\partial \boldsymbol{a}_i)^T + (\partial I_{ji}/\partial \boldsymbol{a}_j)\boldsymbol{P}_{\boldsymbol{a}_j}(\partial I_{ji}/\partial \boldsymbol{a}_j)^T$$
(72)

and we further note that

$$\rho_{ij}\sigma_{I_{ji}}\sigma_{I_{ij}} = \rho_{ij}\sigma_{I_{ij}}\sigma_{I_{ji}} \tag{73}$$

Thus, the equations given in Eqs. (68 - 73) enable one to compute the covariance of a pair of crater pattern invariants when the coefficients and covariances of the respective ellipse fits are known.

RESULTS

Two numerical experiments are conducted to verify the analytical results presented, and the presented techniques are then applied to a series of real-world datasets.

Comparison of Different Ellipse Fit Algorithms

First, we validate the superiority of HLS as compared to LS in the context of ellipse fitting by constructing a simple scenario. While HLS generally provides an unbiased estimate of an ellipse fit, its increased efficacy over LS is especially evident when only a fraction of the locus of points lying on the ellipse are available for fitting. To this end, a simple "truth" ellipse is constructed centered around zero with a semimajor axis length of 0.8, semiminor axis length of 0.6, and a clocking angle of zero. Thirty points are sampled from a quarter of the perimeter, and Gaussian noise is added to both the x- and y-coordinates. Figure 3a shows noise with $\sigma = 0.001$, while Figure 3b shows a more extreme case, with $\sigma = 0.005$. In the nominal noise case, HLS is clearly superior to LS and also demonstrably outperforms Taubin's method.¹⁴ However, in the higher noise case, LS fails to

find a reasonable approximation, and Taubin's method yields an inaccurately large fit, while HLS is able to fit the truth ellipse almost exactly.

In this particular example, the noise levels and set of sampled points are selected in a somewhat arbitrary manner to demonstrate the available performance of HLS. However, subsequent numerical analyses utilize noise values grounded in anticipated sensor characteristics and take advantage of as much of the ellipse perimeter as possible, in order to best match potential operational conditions.



Figure 3 Two examples of ellipse fitting algorithm comparisons. a) 30 points are sampled from a quarter of the truth ellipse perimeter with Gaussian noise of $\sigma = 0.001$ added to both x- and y-coordinates. Three ellipse fitting methods are used to reconstruct the ellipse. b) The same scenario is present, but with Gaussian noise of $\sigma = 0.005$. LS tends to break down in this regime, while HLS is successful in recreating the truth ellipse.

Validation of Analytic Covariance Results

Next, to verify the analytically derived covariance of the invariants, we construct a simulation. The simulation is such that a hypothetical spacecraft "images" two craters of configurable size from a configurable pose.

In order to anchor the simulation in reality, the simulated imager is chosen to have similar camera parameters to the Apollo Metric Camera.¹⁵ These parameters are shown in Table 1. The simulation assumes two craters are visible to the camera, with physical parameters shown in Table 2.

| Parameter | Value | Unit |
|-----------------------|-------|---------------|
| Width | 1024 | pixels |
| Height | 1024 | pixels |
| Pixel Pitch (X and Y) | 9.0 | μ m/pixel |
| Focal Length | 6.0 | mm |

 Table 1
 Simulated Camera Parameters

| Crater | Semimajor | Semiminor | Clocking | X-Distance | Y-Distance | Z-Distance |
|----------|-----------|-----------|----------|------------|------------|------------|
| | Axis [km] | Axis [km] | Angle | From | From | From |
| | | | [deg] | Camera | Camera | Camera |
| | | | | [km] | [km] | [km] |
| Crater 1 | 10 | 7 | 45 | -20.0 | 28.3 | 100 |
| Crater 2 | 8 | 7 | 120 | -17.3 | 28.3 | 100 |

 Table 2
 Simulated Scene Parameters

A Monte Carlo analysis is conducted with 10,000 trials. During each trial, the "truth" crater rim ellipse is projected into the camera frame and 30 random sample points are picked out from around the entire rim. Noise is added to both the x- and y-coordinates of each sampled point with a standard deviation of one pixel,⁵ in line with expected crater detection algorithm noise levels when coupled with subpixel refinement routines. These sampled points are then fit to an ellipse via HLS. Figure 4a shows an overlaid view of the simulated captured images, with Figure 4b providing a detailed view of a single crater, illustrating the envelope of sampled and fitted ellipse rims.



Figure 4 An overlay of Monte Carlo simulated image captures for verifying the analytical covariance formulation for a pair of pattern invariants. a) Full overlaid images captured by the simulation. The nominal crater rims are in black, while each Monte Carlo sampled crater rim is in red. b) A zoomed in view of Crater 2, better illustrating the distribution of crater rims captured in comparison with the nominal (black) rim.

Using these fitted ellipses, crater invariants pairs are calculated, as is their collective simulated covariance. Simultaneously, the parameters of the truth ellipses and the noise parameters of the simulation are used to generate the analytical covariance. An overplot of the resultant 3-sigma error ellipses shown in Figure 5 illustrates that the analytical covariance closely matches the simulation-based covariance.

Note that during the numerical experimentation process, trials with higher noise were explored. In some scenarios, this higher noise excites higher order nonlinearities in HLS, and tends to cause bias

in the measurements. However, within the regime explored in this particular numerical experiment, HLS performs as desired.



Figure 5 Scatterplot of invariant value error from truth from 10000 Monte Carlo trials, with frequency encoded by color. Overplotted are analytically and experimentally derived $3-\sigma$ ellipses.

With the analytical covariance information now available, we can construct a statistical framework to evaluate estimations of the crater invariant pair. The Mahalanobis distance lends itself well to such an application. Let us first represent the difference between these invariant pairs across images as

$$\delta \mathbf{y} = \begin{bmatrix} I_{ij,1} \\ I_{ji,1} \end{bmatrix} - \begin{bmatrix} I_{ij,2} \\ I_{ji,2} \end{bmatrix}$$
(74)

Recalling the calculated invariant covariance formulation from Eq. (68), we may calculate $P_{y_{ij},1}$ and $P_{y_{ij},2}$. The Mahalanobis distance between these invariant pairs is

$$d = \sqrt{\delta \mathbf{y}^T (\mathbf{P}_{y_{ij},1} + \mathbf{P}_{y_{ij},2})^{-1} \delta \mathbf{y}}$$
(75)

The Mahalanobis distance represents a measure of how many standard deviations away the values lie from the mean of their distribution, and its square (that is, d^2) follows a χ^2 distribution.¹⁶ This is verified by overplotting a histogram of d^2 values (relative to truth) from the Monte Carlo simulation with the correspondingly scaled χ^2 distribution, as shown in Figure 6. The inherent relationship between d and a statistical distribution allows for the formulation of a rigorous set of statistical criteria to determine whether the same invariant pair is being viewed across two or more images – a critical step for pose estimation. Discussion of appropriate statistical thresholds is left to the OPNAV practitioner to tune for their specific application.



Figure 6 Plot of the square of Mahalanobis distance of each Monte Carlo trial from truth (histogram bars), overlaid against an appropriately scaled χ^2 distribution (red).

Examples with Flight Data

Figure 1 provided an example of these techniques on real-world Artemis I imagery, examining crater pattern invariants between three pairs of craters. This investigation of real-world data is continued through four subsequent examples—an additional case from Artemis I, and three from Dawn at Ceres—in Figure 7. In each case, craters rim points were manually identified and ellipses were fit using HLS. The inclusion of data from Ceres demonstrates that these techniques are effective beyond lunar applications. Imagery of any body with characteristically elliptical crater rims may be utilized for calculating crater pattern invariants, provided a conventional camera is used, as is the case with the Dawn Framing Camera.¹⁷ Recall that neither the parameters of the cameras used to capture these images nor the current state of the vehicle is necessary to calculate these invariants. Indeed, imagery for the Artemis I examples is obtained from the public NASA/JSC Flickr page,¹⁸ and such metadata is not made available alongside this imagery. In the case of the Ceres examples, the data is gathered from NASA PDS,¹⁹ where pose and camera metadata *is* available, but is not employed for this analysis.

The example shown in Figure 7a is especially indicative of the invariant nature of this technique with respect to pose. Image FC21B0041334_15236174318F2G was captured during the High Altitude Mapping Orbit (HAMO) phase of the Dawn mission at Ceres, with the craters of interest almost directly nadir from the spacecraft. Image FC21B0043185_15267083218F1E was captured during the same HAMO phase, but at a later cycle, from a much more oblique perspective. Despite these differing camera poses, statistically similar (see Table 3) crater pattern invariants emerge.

The examples in Figures 7b and 7c serve to show that this technique is flexible regarding the scale of the craters being examined. In Figure 7b, images FC21B0059418_16070142115F1B and FC21B0063155_16114021252F1B were captured during different cycles of the Low Altitude Mapping Orbit (LAMO) phase of the Dawn mission at Ceres. Despite the significantly smaller crater size (the resolution of LAMO imagery being roughly three times higher compared to HAMO), these



Figure 7 Four real-world examples of crater pattern invariants. Each example contains four images: two raw images showing an area of interest (boxed in red), and the corresponding images directly below showing a zoomed view of these respective areas. In each case, these zoomed views show the same crater rims from different poses. Each crater rim is fitted and crater pattern invariants are calculated. Dawn images are obtained via NASA PDS, while Artemis I images are obtained via the publicly available NASA/JSC Flickr page: https://www.flickr.com/photos/nasa2explore. a) Images FC21B0041334_15236174318F2G (left) and FC21B0043185_15267083218F1E (right) captured during the High Altitude Mapping Orbit phase of Dawn's mission at Ceres. b) Images FC21B0059418_16070142115F1B (left) and FC21B0063155_16114021252F1B (right) captured during the Low Altitude Mapping Orbit phase Dawn. c) Images FC21B0101372_18245071933F1E (left) and FC21B0101370_18245064733F1E (right) captured during the Extended Mission Orbit 7 phase of Dawn. d) Images art001e002596 (left) and art001e002595 (right) captured during Flight Day 20 of the Artemis I mission.

techniques are still effective at producing statistically similar cratern pattern invariants. The images in Figure 7c were captured during the Extended Mission Orbit 7, in which Dawn was placed in a highly elliptical orbit about Ceres. While these images were captured from a higher altitude compared to the HAMO and LAMO images, they demonstrate that—given appropriate crater geometry and visibility, both of which are satisfied in this image pair—these techniques are still applicable.

The Artemis I examples, which span three pairs of craters from a pair of images in Figure 1 and an additional pair of craters from another pair of images in Figure 7d, provide context for the application of crater pattern invariants in a lunar environment. These images demonstrate clearly distinguishable patterns of elliptical craters evident from a typical operational lunar altitude. Crater 2 in Figures 1c and 1d does not contain a full rim of sampled points due to an intersecting crater, a feature common on the lunar surface—however, HLS still yields a good fit, and invariants between this and the other two craters are still able to be calculated.

In all cases, ground truth analytical ellipse rims are strictly unavailable. However, the Mahalanobis distance between invariant pairs calculated from two images targeting the same set of craters still provides a meaningful statistic (χ^2 distributed) for the correctness of these crater invariant calculations. Furthermore, the analytically derived covariance formulation for these invariants directly enables the calculation of such a metric. These distances are recorded in Table 3, and generally indicate that most invariant pairs lie within a single standard deviation of their "expected" values. We note that interpretation of these distances during application of crater pattern invariants in a future mission will largely be a function of the mission and intended application.

| Case | Figure 1, Craters 1 & 2 | Figure 1, Craters 1 & 3 | Figure 1, Craters 2 & 3 | Figure 7a | Figure 7b | Figure 7c | Figure 7d |
|----------------------------|-------------------------------|-------------------------------|-------------------------------|--------------|--------------|--------------|--------------|
| Body | Moon | | | Ceres | | | Moon |
| Mahalanobis Distance, d | 1.066 | 0.776 | 0.988 | 1.318 | 0.686 | 0.824 | 1.188 |

 Table 3
 Real World Crater Pattern Distances

CONCLUSION

Patterns of craters with elliptically shaped rims, when imaged by a conventional camera, result in imaged patterns of ellipses. The mathematical parameters of these ellipses may be combined into values that are invariant to the pose of the imager. Such invariant values may be matched against a precompiled database of invariants corresponding to known crater patterns. This, in turn, may be a valuable tool in a navigation pipeline—as these invariants necessitate no *a priori* knowledge of state, they are ideal for "lost-in-space" scenarios. However, determining the mathematical parameters of the ellipse characterizing a crater rim will necessarily require points of the rim to be sampled and an ellipse fitting algorithm to be executed. This work demonstrated that Hyper Least Squares (HLS) is an attractive ellipse fitting algorithm for this scenario. It produces unbiased estimates of the ellipse fit to the second order (in contrast to traditional Least Squares, an estimator known to be biased for ellipse fitting), while requiring no iteration. Numerically, HLS was shown to produce accurate ellipse fits even in environments where large portions of the ellipse rim points were unavailable, outperforming other non-iterative algorithms. A derivation of HLS was provided, as well as an

analytical formulation of the covariance of resultant ellipse fit. A tradeoff with HLS is slightly increased computational time, though if a large enough sample of points is collected, a simplification may be made, resulting in *Semi*-hyper Least Squares (SHLS), a similar (yet less computationally intensive) algorithm yielding almost identical results in many cases.

Secondly, the noise inherent in this crater selection and ellipse fitting routine was propagated to a measure of covariance for a pair of crater pattern invariants. An analytic expression for the invariant covariance allows for the direct computation of uncertainty in the crater pair invariants. This enables a statistical framework (via a χ^2 test) for rigorously comparing the measured invariants between two images or between an image and a catalog. Real-world applications of these techniques were provided using lunar imagery (from Artemis I) and images of Ceres (from Dawn). Given a lack of ground-truth data in both cases, these examples consisted of two views of the same crater pattern from different poses. A Mahalanobis distance was then calculated between each pair. These applications showed promise, with these distances routinely lying near or below 1.

Any crater pattern invariant scheme will face the effects of noise in real-world applications. Utilizing HLS for the calculation of the crater rim ellipses inherent to the problem and coupling this with a rigorous statistical framework will provide an avenue to handle these effects on potential missions to the Moon and beyond.

ACKNOWLEDGMENT

This work was made possible by NASA under award 80NSSC22M0151. The authors thank Christopher D'Souza, Jorge Chong, Vaishnavi Ramanan, and Liam Smego for valuable discussions on the problem of crater-based lunar navigation.

REFERENCES

- J. Christian and H. Derksen, "Invariant Theory as a Tool for Spacecraft Navigation," AAS/AIAA Astrodynamics Specialist Conference, Paper AAS 22-746, Charlotte, NC, 8 2022.
- [2] J. A. Christian, H. Derksen, and R. Watkins, "Lunar Crater Identification in Digital Images," *The Journal of the Astronautical Sciences*, Vol. 68, 12 2021, pp. 1056–1144.
- [3] W. F. Bottke, S. G. Love, D. Tytell, and T. Glotch, "Interpreting the Elliptical Crater Populations on Mars, Venus, and the Moon," *Icarus*, Vol. 145, 5 2000, pp. 108–121.
- [4] R. R. Herrick, P. M. Schenk, and S. J. Robbins, "Surveys of elliptical crater populations on the saturnian satellites, Mercury, and Mars," *Icarus*, Vol. 220, 8 2012, pp. 297–304.
- [5] D. T. Renshaw and J. A. Christian, "Subpixel Localization of Isolated Edges and Streaks in Digital Images," *Journal of Imaging*, Vol. 6, 5 2020, p. 33.
- [6] A. Fitzgibbon, M. Pilu, and R. B. Fisher, "Direct least square fitting of ellipses," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 21, No. 5, 1999, pp. 476–480.
- [7] K. Kanatani, "Statistical bias of conic fitting and renormalization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 16, 3 1994, pp. 320–326.
- [8] A. Al-Sharadqah and N. Chernov, "Error analysis for circle fitting algorithms," *Electronic Journal of Statistics*, Vol. 3, 2009.
- [9] Z. L. Szpak, W. Chojnacki, and A. v. d. Hengel, "Guaranteed Ellipse Fitting with a Confidence Region and an Uncertainty Measure for Centre, Axes, and Orientation," *Journal of Mathematical Imaging and Vision*, Vol. 52, 6 2015, pp. 173–199.
- [10] W. Chojnacki, M. J. Brooks, and A. Vanel, "On the fitting of surfaces to data with covariances," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 22, 11 2000, pp. 1294–1303.
- [11] K. Kanatani and P. Rangarajan, "Hyper least squares fitting of circles and ellipses," Computational Statistics & Data Analysis, Vol. 55, 6 2011, pp. 2197–2208.
- [12] Z. L. Szpak, W. Chojnacki, and A. v. d. Hengel, "A Comparison of Ellipse Fitting Methods and Implications for Multiple-View Geometry Estimation," 2012 International Conference on Digital Image Computing Techniques and Applications (DICTA), IEEE, 12 2012, pp. 1–8.

- [13] K. B. Petersen and M. S. Pedersen, Matrix Cookbook. 11 2012.
- [14] G. Taubin, "Estimation of planar curves, surfaces, and nonplanar space curves defined by implicit equations with applications to edge and range image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelli*gence, Vol. 13, 1991, pp. 1115–1138.
- [15] K. L. Edmundson, O. Alexandrov, B. A. Archinal, K. J. Becker, T. L. Becker, R. L. Kirk, Z. M. Moratto, A. V. Nefian, J. O. Richie, and M. S. Robinson, "Photogrammetric processing of Apollo 15 metric camera oblique images," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. XLI-B4, 6 2016, pp. 375–381.
- [16] R. G. Brereton, "The chi squared and multinormal distributions," *Journal of Chemometrics*, Vol. 29, 1 2015, pp. 9–12.
- [17] H. Sierks, H. U. Keller, R. Jaumann, H. Michalik, T. Behnke, F. Bubenhagen, I. Büttner, U. Carsenty, U. Christensen, R. Enge, B. Fiethe, P. Gutiérrez Marqués, H. Hartwig, H. Krüger, W. Kühne, T. Maue, S. Mottola, A. Nathues, K. U. Reiche, M. L. Richards, T. Roatsch, S. E. Schröder, I. Szemerey, and M. Tschentscher, "The Dawn framing camera," *Space Science Reviews*, Vol. 163, 12 2011, pp. 263–327.
- [18] NASA Johnson Space Center, "NASA Johnson Flickr," 12 2022. https://www.flickr.com/photos/nasa2explore.
- [19] A. Nathues, H.Sierks, P. Gutierrez-Marques, J. Ripken, I. Hall, I. Buettner, M. Schaefer, and U. Chistensen, "DAWN FC2 RAW (EDR) CERES IMAGES V1.0, DAWN-A-FC2-2-EDR-CERES-IMAGES-V1.0," NASA Planetary Data System, 2015.