



Engineering Notes

Noniterative Horizon-Based Optical Navigation by Cholesky Factorization

John A. Christian*

West Virginia University, Morgantown, West Virginia 26506
and

Shane B. Robinson†

NASA Johnson Space Center, Houston, Texas 77058

DOI: 10.2514/1.G000539

I. Introduction

OPTICAL navigation (OPNAV) using the lit horizon of a planet or moon has received considerable attention over the past few decades for both robotic and human spaceflight. Recently, NASA's Orion Program has investigated the use of horizon-based OPNAV to allow the vehicle to autonomously navigate back to Earth in the event of a communication system failure. This application has created a need for improved performance of horizon-based methods, which has resulted in new work in the past few years [1–9].

Noniterative techniques for horizon-based OPNAV relative to a spherical planet are well known, and they date back to the 1960s [10]. To date, however, the published techniques that provide an exact solution for horizon-based OPNAV relative to an ellipsoidal planet require iteration. This work leverages the Cholesky factorization to produce a noniterative solution for horizon-based OPNAV relative to a triaxial ellipsoidal planet that is exact, and it may be implemented in only a few lines of code, making it an attractive approach for implementation in navigation software on board a spacecraft. The simplicity and geometric clarity of the final result, coupled with its applicability to a triaxial ellipsoidal planet of arbitrary shape, represents a significant step forward in the state of the art for horizon-based OPNAV.

There are a variety of OPNAV techniques that use images of planets against a star field background. The most common variants include star-landmark angles, star-horizon angles, and whole limb fitting. All three methods have found use in past missions. Horizon-based methods for spacecraft navigation were first explored as an experiment during the Gemini program [11], were later demonstrated during the Apollo program [12–14], and reached operational maturity in planetary exploration missions after the Voyager 1 flyby of Jupiter [15]. The use of images for OPNAV has been ubiquitous for precise navigation during planetary encounters since the 1980s [16,17].

In addition to the aforementioned three methods that have found widespread use in past spaceflight missions, there is often other usable information in images that may be exploited for spacecraft

navigation. These other vision-based techniques (such as coastline matching [18], autonomous lunar crater pattern matching [19], star occultation [20], and optical flow [21,22]) have seen considerably less flight use to date.

Of the OPNAV methods listed previously, the authors assert that whole limb fitting is best suited for a backup OPNAV system that must robustly operate after a system failure. Whole limb fitting techniques gain significant robustness by using only horizon measurements and not necessarily requiring star observations. The importance of viable OPNAV using only the lit horizon of a planet is supported by the Apollo 13 experience. In postflight debriefs, the crew of Apollo 13 reported that the ability to use the Sun as well as the lit horizons of the Moon and the Earth was extremely valuable as an attitude reference [23]. This was because the debris field created by the explosive failure of the service module's number 2 oxygen tank impaired the crew's ability to correctly identify stars with the needed certainty, and it is reasonable to presume that this difficulty in correctly identifying stars in the presence of debris would be shared by an OPNAV camera.

The objective of horizon-based OPNAV is to estimate the relative position between the spacecraft and the observed planet using only knowledge of the planet's shape and orientation, the camera attitude, and the line-of-sight directions to points along the planet's lit limb. Such line-of-sight measurements are readily obtained from the apparent location of the lit limb in a single image. The spacecraft attitude allows for the mapping of the planet's orientation into the camera frame and for the eventual conversion of the native OPNAV measurement (which is in the camera frame) back into the inertial frame.

The present work provides a simple and noniterative approach to solving a critical problem for future spaceflight missions. The remainder of the Note is organized as follows. First, the geometric fundamentals of the OPNAV problem are reviewed. Second, Cholesky factorization is shown to transform any horizon-based OPNAV problem into a very simple problem. Third, this simplified problem is shown to have a noniterative solution, and the associated covariance may be found analytically. Fourth, the new algorithm is summarized and shown to only require a few lines of code to implement. Fifth, and finally, numerical results are presented to validate the theoretical developments and to compare with previous techniques. These comparisons show effectively equivalent accuracy with a substantial reduction in computational effort.

II. OPNAV Geometry Fundamentals

Horizon-based OPNAV is a conic section problem [4]. If the planet is modeled as a triaxial ellipsoid, then the apparent horizon is formed by the rays emanating from the camera that are tangent to the planet surface. The collection of all of these rays forms an elliptic cone that tightly bounds the planet with its vertex located at the camera optical center. The perspective projection of the horizon is formed by taking a planar slice of this cone with the image plane. Thus, the apparent horizon of an ellipsoidal planet in an image is formed by a conic section, and it is found to be an ellipse in all reasonable OPNAV scenarios. This is all shown graphically in Fig. 1. The fundamental relations that govern OPNAV geometry are now reviewed in more detail.

A. Perspective Projection and the Pinhole Camera Model

Ideal image formation is governed by perspective projection and can be described by the pinhole camera model. Although real cameras are not pinholes and suffer from an assortment of distortions and imperfections, these may be accounted for via calibration to produce a result functionally equivalent to a pinhole camera [24].

Received 22 March 2016; revision received 30 May 2016; accepted for publication 2 June 2016; published online 29 July 2016. Copyright © 2016 by John A. Christian. Published by the American Institute of Aeronautics and Astronautics, Inc., with permission. Copies of this paper may be made for personal and internal use, on condition that the copier pay the per-copy fee to the Copyright Clearance Center (CCC). All requests for copying and permission to reprint should be submitted to CCC at www.copyright.com; employ the ISSN 0731-5090 (print) or 1533-3884 (online) to initiate your request.

*Assistant Professor, Department of Mechanical and Aerospace Engineering, Benjamin M. Statler College of Engineering and Mineral Resources, Senior Member AIAA.

†Aerospace Engineer, GN&C Autonomous Flight Systems Branch.

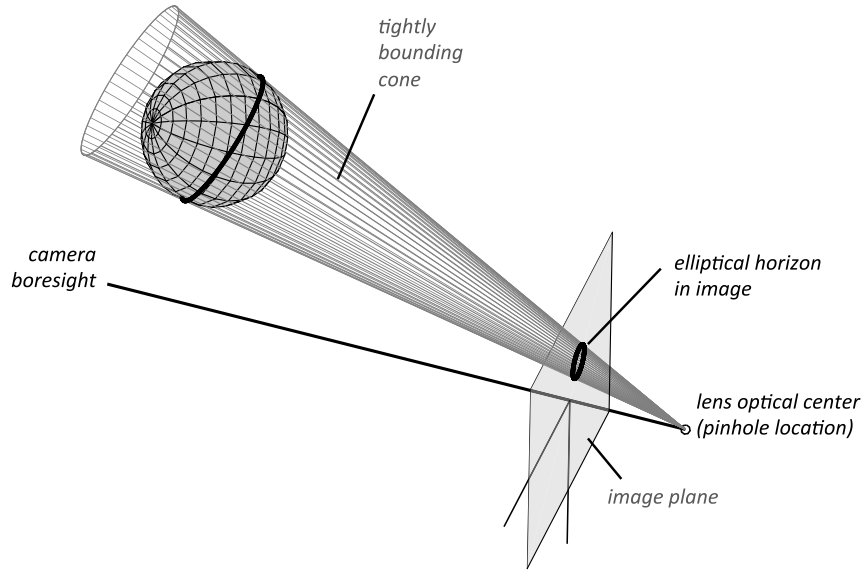


Fig. 1 Geometry of horizon-based OPNAV as a conic section.

Therefore, proceed with the standard description of the pinhole camera [25]:

$$x = f \frac{X}{Z} \quad y = f \frac{Y}{Z} \quad (1)$$

where f is the focal length, $[X, Y, Z]$ are a point's three-dimensional (3-D) coordinates in the camera frame, and $[x, y]$ are the coordinates of the perspective projection of that point onto the image plane. The camera frame is defined to have its origin at the camera's optical center (the pinhole location) with $+Z$ pointing along the boresight of the camera. Thus, any point observed by the camera will have a positive Z component. All operations are performed on the image plane (instead of on the focal plane) to avoid unnecessarily dealing with a flipped image.

Without loss of generality, one may choose to work in focal-length normalized coordinates and simply set $f = 1$ to arrive at

$$x = X/Z \quad y = Y/Z \quad (2)$$

The image plane coordinates may also be written in terms of homogeneous coordinates:

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \frac{1}{Z} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad (3)$$

which makes it apparent that the image plane coordinates occur at the place where the vector from the camera frame origin to the 3-D point pierces the image plane (which is just the $Z = 1$ plane). Conversely, any point in an image may be used to form a ray that starts at the origin, passes through the point in the image plane, and then emanates out of the camera in this direction.

B. Apparent Horizon Location Formed by a Conic Section

Begin by assuming that the planet or moon to be used for OPNAV may be reasonably modeled as a smooth triaxial ellipsoid (the term "smooth" implies that "roughness" due to terrain is neglected). The surface of a triaxial ellipsoid in its own principal axis frame is described by

$$\frac{x_P^2}{a^2} + \frac{y_P^2}{b^2} + \frac{z_P^2}{c^2} = 1 \quad (4)$$

where $\mathbf{p}_P^T = [x_P \ y_P \ z_P]$ is a point on the planet's surface, and $\{a, b, c\}$ are the lengths of the planet's principal axes. The subscript P

denotes the planet's principal axis frame. The preceding may be rewritten as the quadratic form

$$\mathbf{p}_P^T \mathbf{A}_P \mathbf{p}_P = 1 \quad (5)$$

where

$$\mathbf{A}_P = \begin{bmatrix} 1/a^2 & 0 & 0 \\ 0 & 1/b^2 & 0 \\ 0 & 0 & 1/c^2 \end{bmatrix} \quad (6)$$

For the purposes of OPNAV, it is often more desirable to describe the planet shape in a different frame, for which the axes are aligned with the camera frame. Thus, defining \mathbf{T}_P^C as the rotation matrix from the camera frame to the planet's principal axis frame, observe that

$$\mathbf{p}_P = \mathbf{T}_P^C \mathbf{p}_C \quad (7)$$

$$\mathbf{p}_C^T \mathbf{T}_C^P \mathbf{A}_P \mathbf{T}_P^C \mathbf{p}_C = 1 \quad (8)$$

where \mathbf{p}_C is a vector from the planet center to the planet surface as expressed in the camera frame. Now, letting

$$\mathbf{A}_C = \mathbf{T}_C^P \mathbf{A}_P \mathbf{T}_P^C \quad (9)$$

observe that

$$\mathbf{p}_C^T \mathbf{A}_C \mathbf{p}_C = 1 \quad (10)$$

where \mathbf{A}_C is the 3×3 symmetric positive-definite matrix describing the planet's shape in the camera frame. From this point forward, all work will be in the camera frame and the C subscript will be dropped:

$$\mathbf{p}^T \mathbf{A} \mathbf{p} = 1 \quad (11)$$

With such a planet model, the points in the image belonging to the apparent horizon correspond to rays that start at the camera frame origin, emanate out of the camera, and are exactly tangent to the ellipsoidal planet's surface. This family of rays forms a conical surface (again, see Fig. 1). Consequently, the apparent location of the horizon in an image is formed by taking a slice of this cone with the $Z = 1$ plane and is a conic section problem.

Now, suppose a camera views an ellipsoidal planet from a relative position of \mathbf{r} (as expressed in the camera frame). Any particular ray \mathbf{s}_i belonging to the cone that tightly bounds the smooth triaxial ellipsoidal planet obeys the constraint [1,3,4]

$$\mathbf{s}_i^T [\mathbf{A} \mathbf{r} \mathbf{r}^T \mathbf{A} - (\mathbf{r}^T \mathbf{A} \mathbf{r} - 1) \mathbf{A}] \mathbf{s}_i = 0 \quad (12)$$

or

$$\mathbf{s}_i^T \mathbf{M} \mathbf{s}_i = 0 \quad (13)$$

where \mathbf{M} is a symmetric matrix of full rank given by

$$\mathbf{M} = \mathbf{A} \mathbf{r} \mathbf{r}^T \mathbf{A} - (\mathbf{r}^T \mathbf{A} \mathbf{r} - 1) \mathbf{A} \quad (14)$$

Choosing the vector \mathbf{s}_i to describe a point on the image plane, one arrives at $\mathbf{s}_i^T = [x_i \ y_i \ 1]$. Substituting this into Eq. (12) and expanding, one arrives at the quadratic equation

$$\mathbf{s}_i^T \mathbf{M} \mathbf{s}_i = m_{11}x_i^2 + 2m_{12}x_iy_i + 2m_{13}x_i + m_{22}y_i^2 + 2m_{23}y_i + m_{33} = 0 \quad (15)$$

where m_{ij} describes the $[i, j]$ element of the symmetric matrix \mathbf{M} . Observe that this is simply the general implicit equation for any conic section. It is discussed in [4] that the specific conic section will always be an ellipse at the ranges where horizon-based OPNAV is viable (parabolic and hyperbolic horizon arcs are possible at very close range where horizon-based OPNAV techniques are less desirable than landmark-based OPNAV techniques).

III. Cholesky Factorization Space

Although the problem geometry is straightforward for an ellipsoidal planet, it becomes especially intuitive for a spherical planet of unit radius. As will be shown, Cholesky factorization provides a mechanism to transform horizon-based OPNAV relative to an arbitrary triaxial ellipsoidal planet into an idealized unit-sphere geometry. This approach allows any solution or insight available in the simple unit-sphere problem to be applied to the much broader set of all ellipsoidal geometries. These facts will now be shown.

Begin with Eq. (12), which is the equation for a cone tightly bounding the ellipsoid, shown in Fig. 1. Although this equation is insightful, the beauty of its intrinsic geometry is more fully revealed by transforming to a Cholesky factorization space. As will be seen, this transformation provides significant insight and utility. The transformation into Cholesky factorization space is given by

$$\bar{\mathbf{r}} = \mathbf{U} \mathbf{r} \quad \text{and} \quad \bar{\mathbf{s}}_i = \mathbf{U} \mathbf{s}_i \quad (16)$$

where

$$\mathbf{A} = \mathbf{U}^T \mathbf{U} \quad (17)$$

This approach is valid for any transform of the form in Eqs. (16) and (17). For this work, Cholesky factorization is used, where \mathbf{U} is an upper triangular matrix. The authors selected Cholesky factorization for several reasons. First, the Cholesky factorization is well studied [26] and well understood. For Hermitian matrices, the Cholesky factorization is known to exist and be unique. Second, Cholesky factorization routines are ubiquitous in numerical linear algebra software, which provides rapid and stable mechanisms for calculation. Third, the triangular form of the factorization allows the reverse transformation to be carried out in an efficient and stable manner via backward substitution rather than matrix inversion. This is an important consideration for potential implementation on a future spaceflight mission.

Applying the Cholesky factorization to Eq. (12) and using the definitions from Eq. (16) give

$$\bar{\mathbf{s}}_i^T [\mathbf{I}_{3 \times 3} \bar{\mathbf{r}} \bar{\mathbf{r}}^T \mathbf{I}_{3 \times 3} - (\bar{\mathbf{r}}^T \mathbf{I}_{3 \times 3} \bar{\mathbf{r}} - 1) \mathbf{I}_{3 \times 3}] \bar{\mathbf{s}}_i = 0 \quad (18)$$

$$\bar{\mathbf{s}}_i^T \bar{\mathbf{r}} \bar{\mathbf{r}}^T \bar{\mathbf{s}}_i - (\bar{\mathbf{r}}^T \bar{\mathbf{r}} - 1) \bar{\mathbf{s}}_i^T \bar{\mathbf{s}}_i = 0 \quad (19)$$

Observe that Eq. (18) has the exact same form as Eq. (12), but with variables in the Cholesky factorized space ($\bar{\mathbf{r}}$ and $\bar{\mathbf{s}}_i$) and with a planet shape matrix of $\mathbf{I}_{3 \times 3}$. Further observe from Eqs. (6) and (9) that a planet shape matrix of $\bar{\mathbf{A}} = \mathbf{I}_{3 \times 3}$ describes a unit sphere. Thus, the transformation of \mathbf{r} and \mathbf{s}_i into the Cholesky factorized space [Eq. (16)] takes the tightly bounding cone constraint for any arbitrary triaxial ellipsoid [Eq. (12)] and forms an equivalent problem with respect to a unit sphere [Eq. (18)].

Since $\bar{\mathbf{s}}_i$ can be of any magnitude, it is convenient to use a unit vector $\bar{\mathbf{s}}'_i = \bar{\mathbf{s}}_i / \|\bar{\mathbf{s}}_i\|$, which allows Eq. (19) to be rewritten as

$$1 = \bar{\mathbf{r}}^T \bar{\mathbf{r}} - \bar{\mathbf{s}}_i'^T \bar{\mathbf{r}} \bar{\mathbf{r}}^T \bar{\mathbf{s}}_i' \quad (20)$$

This equation is further rearranged to write

$$1 = \bar{\mathbf{r}}^T [\mathbf{I} - \bar{\mathbf{s}}_i' \bar{\mathbf{s}}_i'^T] \bar{\mathbf{r}} \quad (21)$$

where defining $\bar{\mathbf{b}}_i'$ as

$$\bar{\mathbf{b}}_i' = [\mathbf{I} - \bar{\mathbf{s}}_i' \bar{\mathbf{s}}_i'^T] \bar{\mathbf{r}} \quad (22)$$

yields

$$1 = \bar{\mathbf{r}}^T \bar{\mathbf{b}}_i' = \bar{\rho} \cos(\bar{\theta}) \quad (23)$$

where $\bar{\rho} = \|\bar{\mathbf{r}}\|$ is the range in the Cholesky factorized space. The projective geometry shown in Fig. 2 now becomes clear. Note that, in the Cholesky factorized space, $\bar{\theta}$ and $\bar{\phi}$ are the same for all i . Further note that $\bar{\mathbf{b}}_i'$ lies on the unit sphere and is therefore a unit vector: $\|\bar{\mathbf{b}}_i'\| = 1$. To more clearly see that $\bar{\mathbf{b}}_i'$ is a unit vector, observe that $\bar{\mathbf{b}}_i'$ is perpendicular to $\bar{\mathbf{s}}_i'$ [making use of Eq. (22) and that $\|\bar{\mathbf{s}}_i'\| = 1$]:

$$\bar{\mathbf{s}}_i'^T \bar{\mathbf{b}}_i' = \bar{\mathbf{s}}_i'^T [\mathbf{I} - \bar{\mathbf{s}}_i' \bar{\mathbf{s}}_i'^T] \bar{\mathbf{r}} = \bar{\mathbf{s}}_i'^T \bar{\mathbf{r}} - \bar{\mathbf{s}}_i'^T \bar{\mathbf{r}} = 0 \quad (24)$$

This is as expected because $[\mathbf{I} - \bar{\mathbf{s}}_i' \bar{\mathbf{s}}_i'^T]$ is a standard projection matrix (it projects a 3×1 vector onto the plane perpendicular to $\bar{\mathbf{s}}_i'$). Thus, $\bar{\mathbf{b}}_i'$ is the component of $\bar{\mathbf{r}}$ that is perpendicular to $\bar{\mathbf{s}}_i'$, which is simply the vector from the center of the planet to the horizon point in Cholesky factorized space (see Fig. 2). Furthermore, since the planet shape is a unit sphere in Cholesky factorized space, it follows that $\bar{\mathbf{b}}_i'$ must be of unit length.

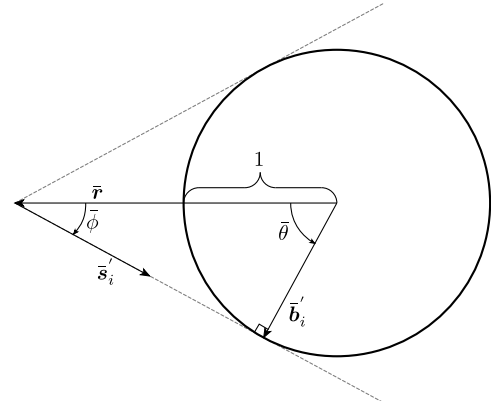


Fig. 2 Geometry of position to limb measurements in Cholesky factorized space.

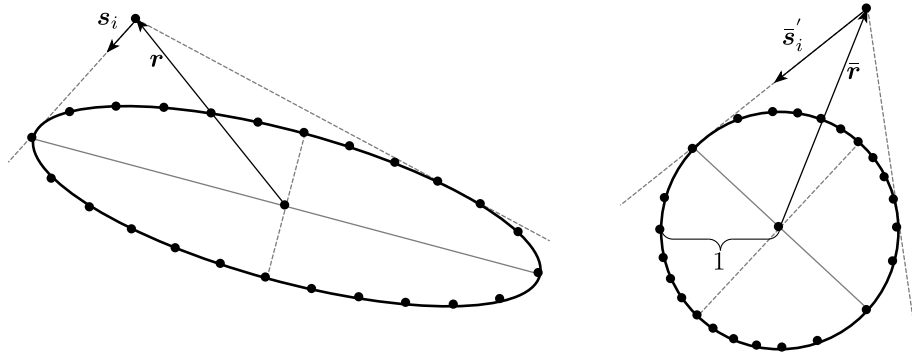


Fig. 3 Comparison of the geometry of the original problem (left) and the same problem in Cholesky factorization space (right).

An alternative perspective is possible by noting that Eq. (21) can be written as

$$1 = \bar{\mathbf{r}}^T [\bar{\mathbf{s}}'_i \times] [\bar{\mathbf{s}}'_i \times] \bar{\mathbf{r}} \quad (25)$$

where $[\cdot \times]$ denotes the skew-symmetric cross-product matrix. This leads directly to two useful relations:

$$\bar{\mathbf{b}}'_i = -\bar{\mathbf{s}}'_i \times (\bar{\mathbf{s}}'_i \times \bar{\mathbf{r}}) \quad (26)$$

$$1 = \|\bar{\mathbf{s}}'_i \times \bar{\mathbf{r}}\| = \bar{\rho} \sin(\bar{\phi}) \quad (27)$$

Again, note that $\bar{\mathbf{b}}'_i$ is easily seen to be a unit vector and that $\bar{\phi}$ is a constant for all $\bar{\mathbf{s}}'_i$ in the Cholesky factorized space.

The geometry associated with Cholesky transformation is shown in Fig. 3. The transformation between the two spaces is a linear transformation defined by \mathbf{U} . This mapping is not isometric. It involves both scaling and rotation, which together cleanly map the ellipsoidal surface to a unit sphere. Note that, when \mathbf{A} is diagonal (such as in the case of an axisymmetric planet), the transformation becomes a pure scaling operation without a rotational component. Although the geometry in Fig. 3 is associated with the Cholesky factorization, other factorizations that satisfy Eq. (17) will produce a similar effect.

It is worth emphasizing the powerful insight and utility provided by viewing the problem in Cholesky factorization space. By providing a transformation from any problem into the simplest problem (i.e., the unit sphere), any relationship available in the unit-sphere problem now becomes applicable to the original problem. This is true even if the original problem involves an arbitrary ellipsoidal planet. Leveraging this newfound insight unlocks a multitude of new approaches to various aspects of the horizon-based OPNAV problem. Perhaps most significantly, it provides a path to a noniterative solution for horizon-based OPNAV relative to an ellipsoidal planet.

IV. Noniterative Horizon-Based OPNAV

As was mentioned earlier, transforming a horizon-based OPNAV problem into Cholesky factorized space turns every problem into navigation relative to a unit sphere. This is true even for very elliptical planets. Fortunately, this simpler geometry admits a noniterative solution.

Since the planet becomes a sphere in the transformed space, its tightly bounding cone is a right-circular cone. Thus, in Cholesky factorized space, the angle between every horizon line-of-sight direction and the relative position vector (which coincides with the cone centerline for a sphere) is the same:

$$\bar{\mathbf{s}}'_i{}^T \bar{\mathbf{r}} = -\|\bar{\mathbf{s}}'_i\| \|\bar{\mathbf{r}}\| \cos(\bar{\phi}) \quad (28)$$

where the negative sign comes from the fact that $\bar{\mathbf{r}}$ goes from the center of the planet to the spacecraft. Define $\mathbf{e} = \bar{\mathbf{r}}/\bar{\rho}$ to be the unit

vector pointing from the planet center to the vehicle in Cholesky factorized space. Therefore, recalling that $\bar{\mathbf{s}}'_i$ is a unit vector, one obtains

$$\bar{\mathbf{s}}'_i{}^T \mathbf{e} = -\cos(\bar{\phi}) \quad (29)$$

Defining the important quantity \mathbf{n} as

$$\mathbf{n} = -\left(\frac{1}{\cos(\bar{\phi})}\right) \mathbf{e} \quad (30)$$

the expression from Eq. (29) may be rewritten to form the following relation for each horizon measurement:

$$\bar{\mathbf{s}}'_i{}^T \mathbf{n} = 1 \quad (31)$$

Figure 4 shows the geometric meaning of \mathbf{n} . The vector \mathbf{n} is a key parameter to the noniterative solution, which is easily estimated by using least squares.

Given a collection of n horizon measurements, \mathbf{n} is seen to be the solution to a simple linear least-squares problem:

$$\mathbf{H} \mathbf{n} = \mathbf{1}_{n \times 1} \quad (32)$$

where

$$\mathbf{H} = \begin{bmatrix} \bar{\mathbf{s}}'_1{}^T \\ \bar{\mathbf{s}}'_2{}^T \\ \vdots \\ \bar{\mathbf{s}}'_n{}^T \end{bmatrix} \quad (33)$$

The vector \mathbf{n} contains all the information necessary to compute the position of the spacecraft relative to the planet in Cholesky factorized space. To see this, simply write the spacecraft position in Cholesky factorized space in terms of direction and magnitude:

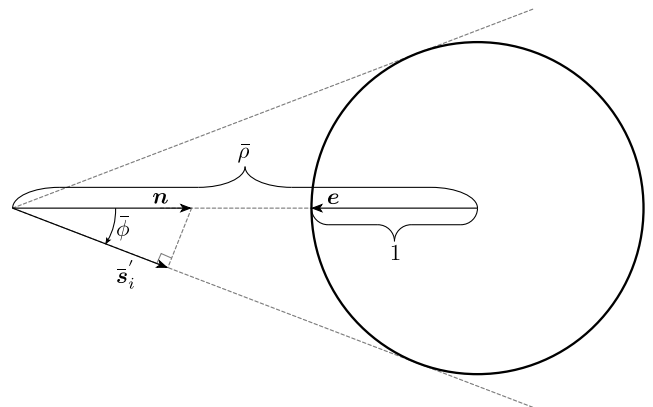


Fig. 4 Geometric meaning of \mathbf{n} .

$$\bar{\mathbf{r}} = \bar{\rho} \mathbf{e} \quad (34)$$

From simple geometry, it becomes clear that

$$\bar{\rho} = 1 / \sin(\bar{\phi}) \quad (35)$$

Note that the $1 / \sin \bar{\phi}$ scaling of range in the context of horizon-based OPNAV relative to a sphere was observed by Battin as early as the 1960s [10]. Thus, the spacecraft position in Cholesky factorized space can be described by just the cone's centerline direction and half-angle:

$$\bar{\mathbf{r}} = \frac{1}{\sin(\bar{\phi})} \mathbf{e} \quad (36)$$

Now, it is straightforward to obtain both \mathbf{e} and $\bar{\phi}$ from the definition of \mathbf{n} in Eq. (30):

$$\mathbf{e} = -\frac{\mathbf{n}}{\|\mathbf{n}\|} = -\frac{\mathbf{n}}{\sqrt{\mathbf{n}^T \mathbf{n}}} \quad (37)$$

$$\|\mathbf{n}\| = \sqrt{\mathbf{n}^T \mathbf{n}} = \frac{1}{\cos(\bar{\phi})} \quad (38)$$

Rather than dealing with the angle $\bar{\phi}$ directly, it is more efficient to use only inner products of \mathbf{n} to compute needed quantities. This use of inner products allows the noniterative solution to be computed without requiring the evaluation of any trigonometric functions. Observing that

$$\mathbf{n}^T \mathbf{n} = \frac{\mathbf{e}^T \mathbf{e}}{\cos^2(\bar{\phi})} = \frac{1}{\cos^2(\bar{\phi})} \quad (39)$$

it is also evident that

$$\sin^2(\bar{\phi}) = 1 - \frac{1}{\mathbf{n}^T \mathbf{n}} \quad (40)$$

Or, perhaps more importantly,

$$\bar{\rho} = \frac{1}{\sin(\bar{\phi})} = \left(\frac{\mathbf{n}^T \mathbf{n}}{\mathbf{n}^T \mathbf{n} - 1} \right)^{1/2} \quad (41)$$

Therefore, substituting Eqs. (37) and (41) into Eq. (34), an estimate for $\bar{\mathbf{r}}$ may be found:

$$\bar{\mathbf{r}} = \bar{\rho} \mathbf{e} = -\left(\frac{\mathbf{n}^T \mathbf{n}}{\mathbf{n}^T \mathbf{n} - 1} \right)^{1/2} \frac{\mathbf{n}}{\sqrt{\mathbf{n}^T \mathbf{n}}} \quad (42)$$

which simplifies to

$$\bar{\mathbf{r}} = -(\mathbf{n}^T \mathbf{n} - 1)^{-(1/2)} \mathbf{n} \quad (43)$$

Finally, $\bar{\mathbf{r}}$ in the Cholesky factorized space may be related to \mathbf{r} in the real world using Eq. (16):

$$\mathbf{U} \mathbf{r} = -(\mathbf{n}^T \mathbf{n} - 1)^{-(1/2)} \mathbf{n} \quad (44)$$

Fortunately, there is no need to compute \mathbf{U}^{-1} to solve for \mathbf{r} . The choice of Cholesky factorization makes \mathbf{U} an upper triangular matrix, so Eq. (44) may be solved for \mathbf{r} in an efficient and numerically stable manner through backward substitution. The solution is trivial for the case where \mathbf{U} is diagonal, which occurs for spherical planets or when the camera frame is aligned with the planet's principal axis frame.

Of course, \mathbf{U} will always be invertible due to the structure of \mathbf{A} and one can write the equation for \mathbf{r} directly as

$$\mathbf{r} = -(\mathbf{n}^T \mathbf{n} - 1)^{-(1/2)} \mathbf{U}^{-1} \mathbf{n} \quad (45)$$

This noniterative solution allows horizon-based OPNAV relative to an arbitrary ellipsoidal planet to be solved using only a small fraction of the computational effort required for the existing iterative solutions, such as from [4]. Furthermore, this solution leads to a convenient representation of the solution covariance.

V. Solution Covariance

In addition to simply generating the OPNAV measurement, it is often important to understand the uncertainty in this measurement. The OPNAV approach introduced in the preceding sections lends itself to a straightforward covariance analysis.

From the least-squares problem in Eq. (32), it is straightforward to find the covariance of \mathbf{n} as

$$\mathbf{P}_n = (\mathbf{H}^T \mathbf{R}_y^{-1} \mathbf{H})^{-1} \quad (46)$$

where \mathbf{R}_y is an $n \times n$ matrix describing the covariance of the residuals:

$$\mathbf{R}_y = \text{diag}[\sigma_{y_1}^2, \sigma_{y_2}^2, \dots, \sigma_{y_n}^2] \quad (47)$$

To make implementation practical, the least-squares residual variances must be related to their corresponding line-of-sight variances. Thus, begin by mapping the actual line-of-sight error into the Cholesky factorization space:

$$\delta \bar{\mathbf{s}}_i = \mathbf{U} \delta \mathbf{s}_i \quad (48)$$

Next, taking the variation of Eq. (31) with respect to $\bar{\mathbf{s}}_i$ (while recalling that $\bar{\mathbf{s}}'_i = \bar{\mathbf{s}}_i / \|\bar{\mathbf{s}}_i\|$), one finds that

$$\delta y_i = \frac{1}{\|\bar{\mathbf{s}}_i\|} \mathbf{n}^T (\mathbf{I}_{3 \times 3} - \bar{\mathbf{s}}'_i \bar{\mathbf{s}}'^T_i) \delta \bar{\mathbf{s}}_i \quad (49)$$

Therefore, defining \mathbf{J} as the 1×3 matrix

$$\mathbf{J}_i = \frac{1}{\|\bar{\mathbf{s}}_i\|} \mathbf{n}^T (\mathbf{I}_{3 \times 3} - \bar{\mathbf{s}}'_i \bar{\mathbf{s}}'^T_i) \quad (50)$$

it is evident that

$$\delta y_i = \mathbf{J}_i \mathbf{U} \delta \mathbf{s}_i \quad (51)$$

Consequently, the variance of the i th residual is

$$\sigma_{y_i}^2 = \mathbf{J}_i \mathbf{U} \mathbf{R}_s \mathbf{U}^T \mathbf{J}_i^T \quad (52)$$

It has previously been shown that the covariance of the horizon measurements, $\mathbf{R}_s = E[\delta \mathbf{s}_i \delta \mathbf{s}_i^T]$, can be well approximated as [4]

$$\mathbf{R}_s \approx \left(\frac{\sigma_{\text{pix}}}{d_x} \right)^2 \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (53)$$

where σ_{pix} is the standard deviation of an observed horizon point in units of pixels, and d_x is the pixel pitch in terms of pixels per radian.

With the covariance of \mathbf{n} known, it is straightforward to find the covariance of \mathbf{r} . Simply take the variation of Eq. (45) with respect to \mathbf{n} :

$$\delta \mathbf{r} = -(\mathbf{n}^T \mathbf{n} - 1)^{-(1/2)} \mathbf{U}^{-1} \left(\mathbf{I}_{3 \times 3} - \frac{\mathbf{n} \mathbf{n}^T}{\mathbf{n}^T \mathbf{n} - 1} \right) \delta \mathbf{n} \quad (54)$$

Thus, defining the 3×3 matrix \mathbf{F} as

$$\mathbf{F} = -(\mathbf{n}^T \mathbf{n} - 1)^{-(1/2)} \mathbf{U}^{-1} \left(\mathbf{I}_{3 \times 3} - \frac{\mathbf{n} \mathbf{n}^T}{\mathbf{n}^T \mathbf{n} - 1} \right) \quad (55)$$

one finds that

$$\delta \mathbf{r} = \mathbf{F} \delta \mathbf{n} \quad (56)$$

At this point, the covariance of \mathbf{r} is easily found to be

$$\mathbf{P}_r = E[\delta \mathbf{r} \delta \mathbf{r}^T] = \mathbf{F} \mathbf{P}_n \mathbf{F}^T \quad (57)$$

Through simple transformations, the OPNAV covariance from Eq. (57) is found to be identical to that first presented by Christian [4] and later parameterized by Hikes and Christian [5] for the iterative horizon reprojecton (IHR) method. Thus, the analytic uncertainty

associated with the noniterative solution in Cholesky factorized space is identical to that of IHR. This is as expected, since both solve the same estimation problem that starts with Eq. (12).

VI. Algorithm Summary

Despite the fascinating geometry and theoretically rich underlying mathematics, the final OPNAV solution is strikingly simple. Indeed, the new horizon-based OPNAV algorithm described in the preceding sections may be implemented in only a few lines of code. It is this simplicity that makes the present technique so attractive for implementation on a flight system.

Of particular note is that computation of the OPNAV measurement requires only five steps. A summary is provided in Fig. 5. Each step is either trivially simple or uses a well-understood numerical technique with extensive flight heritage.

OPNAV($\mathbf{A}, [\mathbf{s}_0, \mathbf{s}_1, \dots, \mathbf{s}_n]$)

- 1) Compute the matrix square root \mathbf{U} using Cholesky factorization.
- 2) Transform the horizon measurements \mathbf{s}_i to Cholesky factorized space with Eq. (16)
- 3) Normalize each $\bar{\mathbf{s}}_i$ to compute $\bar{\mathbf{s}}'_i$ and concatenate them into the matrix \mathbf{H} as in Eq. (33)
- 4) Solve the least-squares problem in Eq. (32) for \mathbf{n} using a robust method, such as QR factorization.
- 5) Compute \mathbf{r} . This is done efficiently by solving Eq. (44) for \mathbf{r} via backward substitution. Alternatively, compute \mathbf{r} directly from Eq. (45)

Fig. 5 Summary of the algorithm to solve the lit planetary horizon OPNAV problem.

OPNAV_COVARIANCE($\mathbf{A}, [\mathbf{s}_0, \mathbf{s}_1, \dots, \mathbf{s}_n], \sigma_{pix}, d_x$)

- 1) Compute \mathbf{r} and \mathbf{n} by following the steps in Fig. 5 to solve the lit planetary horizon OPNAV problem.
- 2) Use Eq. (53) to form \mathbf{R}_s , the covariance of the horizon measurements.
- 3) For each $\bar{\mathbf{s}}'_i$, compute the partial \mathbf{J}_i with Eq. (50)
- 4) For each $\bar{\mathbf{s}}'_i$, compute the variance of each residual, $\sigma_{y_i}^2$, with Eq. (49)
- 5) Use these individual variances to construct the measurement residual covariance, \mathbf{R}_y , using Eq. (47)
- 6) Compute the covariance of the least-square solution, \mathbf{P}_n , using Eq. (46)
- 7) Compute, \mathbf{F} , the partial derivative of \mathbf{r} with respect to \mathbf{n} with Eq. (55)
- 8) Calculate the covariance of the OPNAV solution, \mathbf{P}_r , using Eq. (57)

Fig. 6 Summary of the algorithm to calculate the covariance of the lit planetary horizon OPNAV problem.

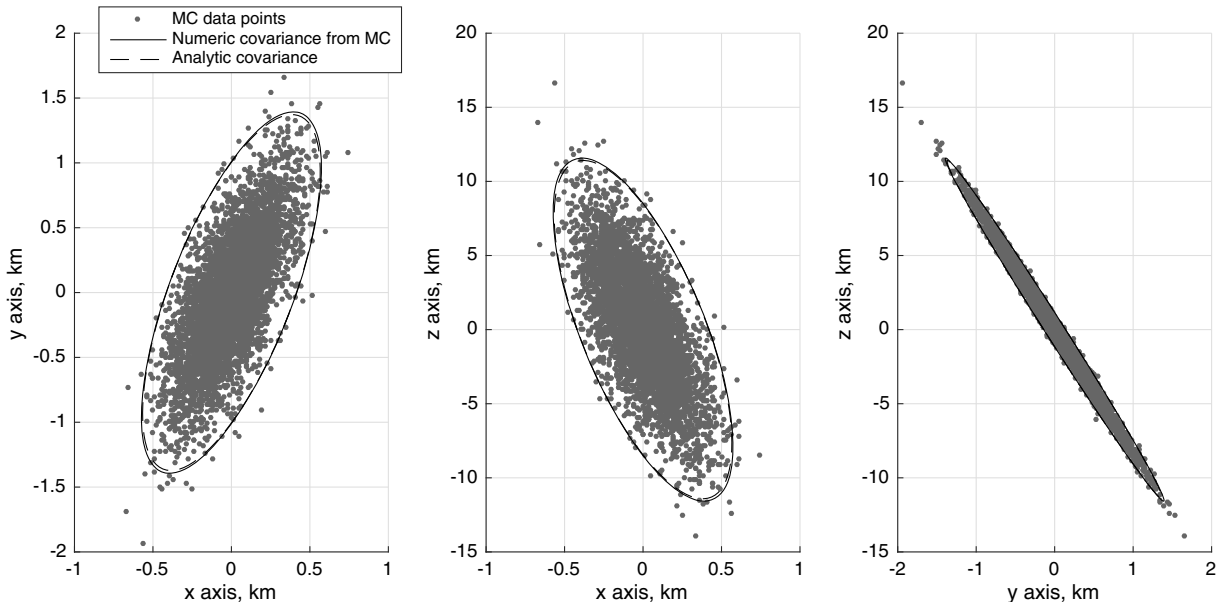


Fig. 7 Monte Carlo (MC) results for the noniterative Cholesky factorization OPNAV technique with respect to a highly elliptical body.

Table 1 Monte Carlo analysis demonstrating the noniterative Cholesky factorization OPNAV measurements are nearly zero mean and approximately Gaussian

Direction	Mean error, km	Standard deviation, km	Skewness	Kurtosis
x axis	−0.0064	0.1910	0.1068	2.9622
y axis	−0.0216	0.4648	0.0331	3.0173
z axis	0.1786	3.8595	−0.0264	3.0344

Further, if the covariance is desired, it may be computed with seven additional steps, as is shown in Fig. 6. As with the OPNAV measurement itself, each step consists of straightforward calculations easily accomplished using existing numerical linear algebra libraries.

VII. Numerical Results

A. Performance Assessment via Monte Carlo Analysis

The performance of the new noniterative Cholesky factorization method was assessed via a Monte Carlo analysis. The results of this



Fig. 8 Example image of the Moon used to generate OPNAV measurement.

analysis were found to agree with the analytic covariance developed in Sec. V. To stress the algorithm and to highlight its effectiveness for highly elliptical bodies, this simulation considered a fictional planet with a major axis radius of 2000 km, an intermediate axis radius of 1500 km, and a minor axis radius of 1000 km. Such a body is substantially more elliptical than most planets or moons that were reasonably modeled as a triaxial ellipsoid.

Given this highly elliptical body, synthetic images and corresponding horizon measurements were generated at a range of about 15,000 km. This process was repeated 5000 times with different measurement noise present in each case. The results are as shown graphically in Fig. 7 and numerically in Table 1. The resulting measurements were found to be very nearly zero mean, to have near-zero skewness, and to have approximately Gaussian kurtosis. These desirable properties are very encouraging if these measurements are to be included in a Kalman filter or other estimation framework.

B. Comparison with Earlier Methods Along a Lunar Return Reference Trajectory

The performance of the noniterative OPNAV algorithm presented in this Note was evaluated along a reference lunar return trajectory with 322 simulated images of the Moon. The images were generated using the Engineering Dynamic On-Board Ubiquitous Graphics for Exploration (EDGE) software package. To make comparisons with earlier work easier, the reference return trajectory and lunar image set used here are exactly the same as in [4]. This dataset includes images similar to that shown in Fig. 8.

The errors in the noniterative OPNAV measurements for this dataset are compared with the errors in the iterative horizon reprojection measurements from [4]. The results of this comparison are shown in Fig. 9. It is immediately evident that the OPNAV measurements generated by the two methods are effectively identical. This is expected because they both solve the same minimization problem (that is, they both yield the exact solution to the perspective projection of a planetary horizon onto an image). The small differences are due to issues with numerical precision and the convergence tolerance used to terminate the IHR iterations.

Additionally, it is observed that the new noniterative method is much faster than the older IHR method for any reasonable number of horizon points. This is mainly because 1) the need to solve an eigenvalue problem is eliminated and 2) because no iterations are required. A comparison of the execution times are shown in Fig. 10.

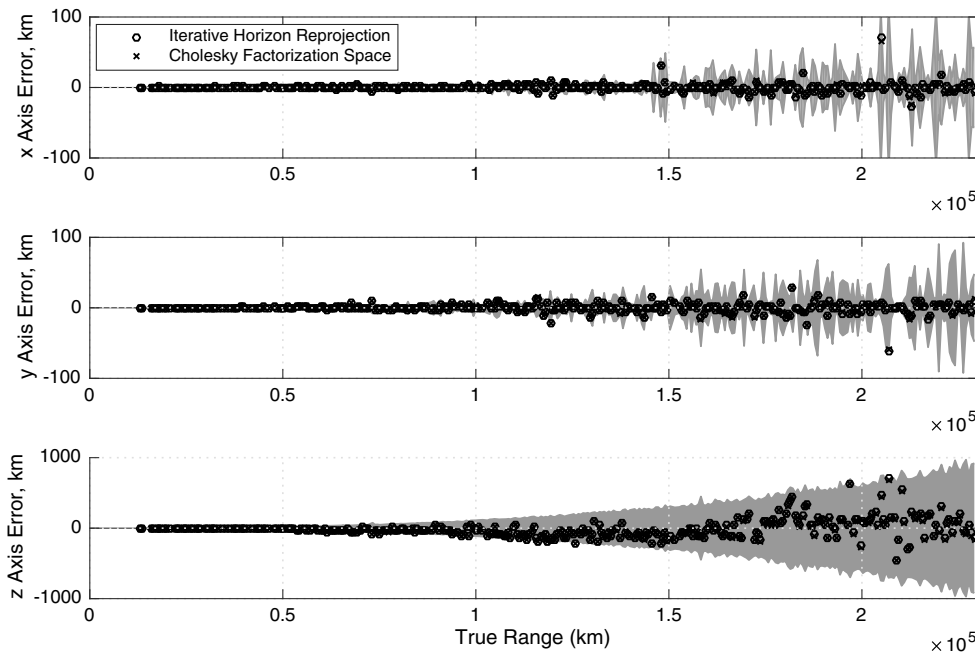


Fig. 9 OPNAV measurement errors for both the IHR method and the noniterative Cholesky method. Shaded gray region is the analytic 3 σ covariance.

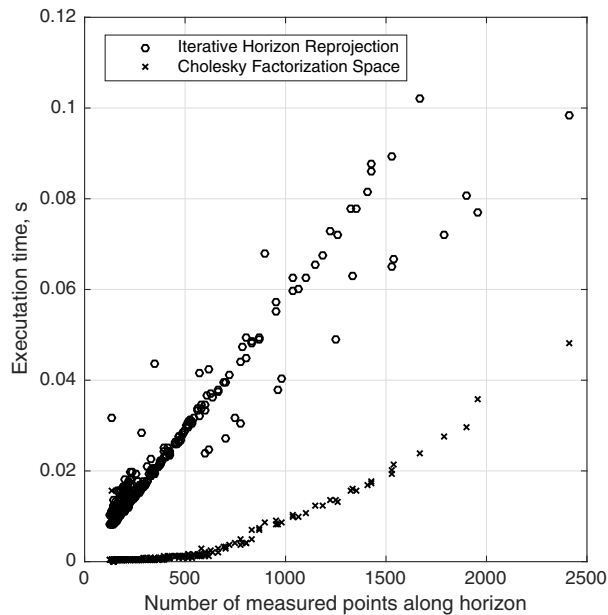


Fig. 10 Execution time for the noniterative Cholesky method, shown substantially faster than the IHR method.

These execution times correspond to the algorithms as implemented by the authors in MATLAB R2015b and run on a MacBook Pro laptop computer with operating system 10.11.1.

The three families of linear slopes for the IHR execution times come from the various numbers of iterations required for convergence (either two, three, or four iterations); this behavior and other observations related to IHR execution time are discussed at length in [4]. For the Cholesky space method, the execution time is dominated by solving the least-square problem in Eq. (32). The authors chose to solve this via QR factorization, and the execution times shown in Fig. 10 reflect this choice. The computational complexity of the QR factorization is $O(n^3)$, so the nonlinear increase in execution time with the number of measured horizon points is expected. This substantial increase in computational efficiency makes this method amenable to use in onboard software, which must execute in near-real time with minimal computational resources.

VIII. Conclusions

This Note explores how key geometric insights provided by Cholesky factorization lead directly to a noniterative solution for horizon-based optical navigation relative to an arbitrary triaxial ellipsoidal planet or moon. It is shown that horizon-based OPNAV is a conic section problem and that application of the Cholesky factorization constitutes a novel method for transforming all problems into the simple case of navigation relative to a unit sphere. In addition to the noniterative solution, this transformation unlocks a clean formulation of the covariance associated with the solution. The performance is shown to be identical to the result obtained by the latest iterative methods, and the Monte Carlo results match the predicted covariance exceedingly well. Finally, it is shown that this new approach requires minimal computational resources when compared with the most advanced iterative methods in the literature. As a result, the approach presented here represents a substantial advancement in the fundamental theory and application of horizon-based OPNAV.

Acknowledgments

John Christian's portion of this work was made possible by NASA under award NNX13AJ25A. The authors thank Chris D'Souza and John Goodman for many insightful discussions that helped improve the quality of this manuscript. The authors would also like to thank

Jim and Danny Roberts for helping view the problem from a new perspective. An earlier version of this Note was presented as American Astronautical Society (AAS) Paper 16-151 at the 39th Annual AAS Guidance, Navigation, and Control Conference in Breckenridge, Colorado, 5-10 February 2016.

References

- [1] Christian, J., "Optical Navigation for a Spacecraft in a Planetary System," Ph.D. Thesis, Univ. of Texas at Austin, Austin, TX, 2010.
- [2] Christian, J., and Lightsey, E., "Onboard Image-Processing Algorithm for a Spacecraft Optical Navigation Sensors System," *Journal of Guidance, Control, and Dynamics*, Vol. 49, No. 2, 2012, pp. 337-352. doi:10.2514/1.A32065
- [3] Christian, J., "Optical Navigation Using Planet's Centroid and Apparent Diameter in Image," *Journal of Guidance, Control, and Dynamics*, Vol. 38, No. 2, 2015, pp. 192-204. doi:10.2514/1.G000872
- [4] Christian, J. A., "Optical Navigation Using Iterative Horizon Reprojection," *Journal of Guidance, Control, and Dynamics*, Vol. 39, No. 5, 2016, pp. 1092-1103. doi:10.2514/1.G001569
- [5] Hikes, J., and Christian, J. A., "Parametric Covariance Model for Horizon-Based Optical Navigation," *AAS Space Flight Mechanics Meeting*, American Astronautical Soc. Paper 16-441, Feb. 2016.
- [6] Borissov, S., and Mortari, D., "Optimal Single-Point and Filtered Pose Estimation for Lunar Orbiters Using Visible Camera," *24th AAS/AIAA Space Flight Mechanics Meeting*, American Astronautical Soc. Paper 14-247, Jan. 2014.
- [7] Mortari, D., de Dilectis, F., and Zanetti, R., "Position Estimation Using the Image Derivative," *Aerospace*, Vol. 2, No. 3, 2015, pp. 435-460. doi:10.3390/aerospace2030435
- [8] Mortari, D., and Borissov, S., "Moon and Earth Image Processing Using Asymmetric 2-D Gaussian Function on Image Gradient," *AAS/AIAA Space Flight Mechanics Meeting*, American Astronautical Soc. Paper 2016-341, 2016.
- [9] Mortari, D., D'Souza, C., and Zanetti, R., "Image Processing of Illuminated Ellipsoid," *Journal of Spacecraft and Rockets*, Vol. 53, No. 3, 2016, pp. 448-456. doi:10.2514/1.A33342
- [10] Battin, R. H., *Astronautical Guidance*, McGraw-Hill, New York, 1964, p. 225.
- [11] Smith, D., and Lampkin, B., "Sextant Sighting Measurements from On Board the Gemini XII Spacecraft," NASA TR TN-D-4952, 1968.
- [12] Draper, C., Wrigley, W., Hoag, G., Battin, R., Miller, J., Koso, D., Hopkins, A., and Vander Velde, W., "Space Navigation Guidance and Control, Vol. 1, of 2," NASA TR NASA-CR-75543, 1965.
- [13] Hoag, D., "The History of Apollo On-Board Guidance, Navigation, and Control," *Journal of Guidance, Control, and Dynamics*, Vol. 6, No. 1, 1983, pp. 4-13. doi:10.2514/3.19795
- [14] Kachmar, P., and Wood, L., "Space Navigation Applications," *Navigation: Journal of the Institute of Navigation*, Vol. 42, No. 1, 1995, pp. 186-234. doi:10.1002/navi.1995.42.issue-1
- [15] Campbell, J., Synnott, S., and Bierman, G., "Voyager Orbit Determination at Jupiter," *IEEE Transactions on Automatic Control*, Vol. 28, No. 3, 1981, pp. 256-268. doi:10.1109/TAC.1983.1103223
- [16] Owen, W., Duxbury, T., Action, C., Synnott, S., Riedel, J., and Bhaskaran, S., "A Brief History of Optical Navigation at JPL," *AAS Guidance and Control Conference*, American Astronautical Soc. Paper 2008-053, 2008.
- [17] Owen, W., "Methods of Optical Navigation," *AAS/AIAA Space Flight Mechanics Meeting*, American Astronautical Soc. Paper 2011-215, Feb. 2011.
- [18] Straub, M., and Christian, J., "Autonomous Optical Navigation for Earth-Observing Satellites Using Coastline Matching," *AIAA Guidance, Navigation, and Control Conference*, AIAA Paper 2015-1334, 2015.
- [19] Hanak, F., "Lost in Low Lunar Orbit Crater Pattern Detection and Identification," Ph.D. Thesis, Univ. of Texas at Austin, Austin, TX, 2009.
- [20] Psiaki, M., and Hinks, J., "Autonomous Lunar Orbit Determination Using Star Occultation Measurements," *AIAA Guidance, Navigation, and Control Conference*, AIAA Paper 2007-6657, Aug. 2007.
- [21] Gurfil, P., and Rotstein, H., "Partial Aircraft State Estimation from Visual Motion Using the Subspace Constraints Approach," *Journal of*

- Guidance, Control, and Dynamics*, Vol. 24, No. 5, 2016, pp. 1016–1028.
doi:10.2514/2.4811
- [22] Cheng, Y., Goguen, J., Johnson, A., Leger, C., Matthies, L., San Martin, M., and Wilson, R., “The Mars Exploration Rovers Descent Image Motion Estimation System,” *IEEE Intelligent Systems*, Vol. 19, No. 3, May–June 2004, pp. 13–21.
doi:10.1109/MIS.2004.18
- [23] Goodman, J. L., “Apollo 13 Guidance, Navigation, and Control Challenges,” *AIAA Space Conference and Exposition*, AIAA Paper 2009-6455, Sept. 2009.
- [24] Christian, J. A., Hikes, J., and Benhacine, L., “Geometric Calibration of the Orion Optical Navigation Camera Using Star Field Images,” *AAS Guidance, Navigation, and Control Conference*, American Astronautical Soc. Paper 2016-116, Feb. 2016.
- [25] Ma, Y., Soatto, S., Koščeká, J., and Sastry, S., *An Invitation to 3-D Vision: From Images to Geometric Models*, Springer, New York, 2010, pp. 49–59.
- [26] Golub, G. H., and Van Loan, C. F., *Matrix Computations*, 4th ed., Johns Hopkins Univ. Press, Baltimore, MD, 2012, pp. 163–164.