IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received December 30, 2020, accepted January 10, 2021, date of publication January 18, 2021, date of current version February 3, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3051914

A Tutorial on Horizon-Based Optical Navigation and Attitude Determination With Space Imaging Systems

JOHN A. CHRISTIAN
Department of Mechanical, Aerospace, and Nuclear Engineering, Rensselaer Polytechnic Institute, Troy, NY 12180, USA
e-mail: chrisj9@rpi.edu

This work was supported in part by the U.S. National Aeronautics and Space Administration (NASA) under Award 80NSSC17M0027.

ABSTRACT Images of a nearby celestial body collected by a camera on an exploration spacecraft contain a wealth of actionable information. This work considers how the apparent location of the observed body's horizon in a digital image may be used to infer the relative position, attitude, or both. When the celestial body is a sphere, spheroid, or ellipsoid (as is the case for most large bodies in the Solar System), the projected horizon in an image is a conic—usually an ellipse at large distances and a hyperbola at small distances. This work develops non-iterative and analytically exact methods for every case (all combinations of unknown state parameters and quadric shapes), completely superseding older horizon-based methods that are iterative, approximate, or both. Some of the analytic methods presented in this work are new. Recognizing that these developments build on techniques that may be unfamiliar to many spacecraft navigators, this work is fashioned as a tutorial. Descriptive illustrations and numerical examples are provided to make concepts clear and to validate the proposed algorithms.

INDEX TERMS Algebraic geometry, attitude determination, computer vision, Earth horizon sensors, optical navigation (OPNAV), space exploration, quadrics, spacecraft navigation.

I. INTRODUCTION

Digital cameras are versatile sensors that are capable of simultaneously supporting both scientific and engineering objectives for space exploration missions [1]. Of special note, the use of images from these cameras for the purpose of spacecraft navigation—often referred to as optical navigation (OPNAV)—has long been an important technology for spacecraft venturing beyond low Earth orbit (LEO) [2].

There are a variety of forms of OPNAV, including those using the horizon (lit limb) of a celestial body [3]–[5], specific features on the body's surface (e.g., craters [6]–[8] or other known landmarks [9]–[12]), and visual odometry [13]. This work focuses exclusively on the problem of horizon-based OPNAV.

We observe that the horizon-based OPNAV problem is tightly linked to the horizon-based attitude determination problem. Although observations of Earth's horizon in the infrared have long been used for spacecraft attitude determination using Earth horizon sensors (both scanning and staring) [14]–[17], the relationship between OPNAV and attitude determination is rarely made. Indeed, the mathematics used to solve these two horizon-based navigation problems have been developed separately—often bearing little obvious connection. This work provides the first known unification of these two problems in a comprehensive manner.

An image of a nearby celestial body provides information about the relative position and relative attitude (sometimes called *pose*) between the spacecraft and the observed celestial body. In addition to the navigation camera, knowledge of position and attitude can also come from a variety of different sources. Moreover, since translational and rotational state estimation is sometimes separated, it is not unusual to know one but not the other. This leads to three natural state estimation scenarios:

1) The relative attitude is known from other sources, but the relative position is unknown. This leads to the classic celestial OPNAV problem where images are used to estimate relative position.

The associate editor coordinating the review of this manuscript and approving it for publication was Ze Ji.

2) The relative position is known from other sources, but the relative attitude is unknown. This leads to two sub-problems, depending on the frame in which the position is known:
   a) Relative position is known in a frame fixed to the celestial body. This implies the celestial body's attitude is known and the spacecraft attitude is unknown (e.g., attitude determination with an Earth horizon sensor). Images are used to estimate the spacecraft attitude.
   b) Relative position is known in the camera frame. This implies the spacecraft attitude is known and the celestial body attitude is unknown (e.g., imaging a celestial body with a well-known orbit but unpredictable attitude such as the Saturnian moon Hyperion [18]). Images are used to estimate the celestial body's attitude.
3) There is no prior relative state knowledge, and both the relative attitude and relative position are unknown. Images are used to estimate the pose between the celestial body and camera.

Most large celestial bodies within the Solar System may be modeled as spheres, spheroids, or triaxial ellipsoids. This leads to a variety of algorithms for handling the different state estimation scenarios when viewing celestial bodies of different shape. The possibilities are summarized in Table 1 along with cross-references to the specific algorithms to solve each case.

TABLE 1. Summary of the different state estimation scenarios when imaging celestial bodies of different shape. The scenarios are grouped by what state information is to be estimated (i.e., is unknown) and what state information is known. The possible relative states used here include: relative position expressed in the camera frame (r_C), relative position expressed in the celestial body's frame (r_P), relative attitude between the celestial body's frame and the camera frame (T_C^P).

Scenario	Section	Algorithm
1) Estimate r_C; Know T_C^P		
Sphere	Sec. III-C	Alg. 1
Spheroid	Sec. III-C	Alg. 1
Triaxial Ellipsoid	Sec. III-C	Alg. 1
2.a) Estimate T_C^P; Know r_P		
Sphere	Sec. IV-A	Alg. 1
Spheroid	Sec. IV-A	Alg. 2
Triaxial Ellipsoid	Sec. IV-A	Alg. 2
2.b) Estimate T_C^P; Know r_C		
Sphere	Sec. IV-B	N/A
Spheroid	Sec. IV-B	Alg. 3
Triaxial Ellipsoid	Sec. IV-B	Alg. 4
3) Estimate r_C & T_C^P; Know nothing		
Sphere	Sec. V-C	Alg. 1
Spheroid	Sec. V-C	Alg. 5
Triaxial Ellipsoid	Sec. V-C	Alg. 6

This manuscript is fashioned as a tutorial. Such a presentation is motivated by widespread confusion in the contemporary literature about the geometry of horizon-based navigation and the state-of-the-art in extracting navigation information from images. The intent is to present an accessible, comprehensive, and self-consistent description of the mathematics necessary to solve these commonly encountered navigation problems. Therefore, after developing the requisite preliminary tools in Sec. II, we present solutions for every horizon-based navigation scenario listed in Table 1.

II. GEOMETRIC PRELIMINARIES

A. MODEL FOR A PROJECTIVE CAMERA

Consider a conventional camera system on a spacecraft that is used to capture an image of a nearby celestial body. This work focuses on projective cameras that produce a 2D digital image, sometimes called a *framing camera* in the space imaging community (as opposed to a linescan or pushbroom camera [19]). A typical framing camera suitable for space imaging consists of a few system-level components as shown in Fig. 1.

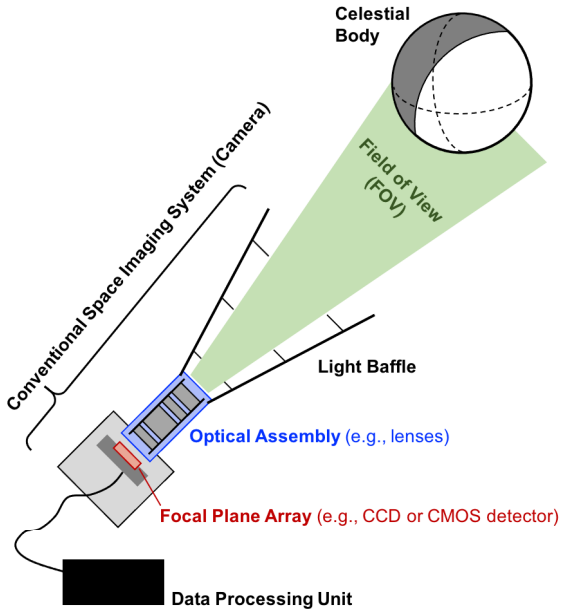

FIGURE 1. Major system-level components of a conventional framing camera system viewing a celestial body.

Light baffles are used to block stray light from striking the camera aperture and are an essential component of space imaging systems [20], [21]. Photons arriving from directions within the camera field-of-view (FOV) are allowed to pass by the light baffle, are collected by the aperture of the optical assembly, and are focused onto a detector residing on the focal plane. Multi-lens refractive optical assemblies, like the one implied in Fig. 1, generally introduce a number of optical aberrations [22]—with radial distortion being among the most important for OPNAV applications. Additional geometric distortions in the projected image are caused by manufacturing imperfections and component misalignment. Fortunately, these effects can be mostly removed with straightforward geometric calibration techniques [23]. The details are not discussed here (see [23] and [24] for practical examples), but starfield-based calibration can produce corrected images that are nearly indiscernible from perfect

perspective projection (i.e., the so-called pinhole camera model). The remainder of this article assumes that optical distortions are removed and images are formed by perfect perspective projection.

In the physical camera system, the detector lies inside the camera housing on (or near) the focal plane of the optical system. Older camera systems (e.g., the Apollo mapping/metric camera [25]) used film to record images, whereas contemporary systems create a digital image with a 2D array of photodetectors (e.g., CCD and CMOS imaging sensors [26]). We generically refer to these detectors as focal plane arrays (FPAs). A 2D digital image is formed by placing the measured brightness from each photodetector in the FPA into the corresponding image pixel.

In an effort to directly model the image formation process, early work in image-based spacecraft navigation converted the image pixel coordinates to their physical locations on the focal plane. It soon became apparent, however, that such an approach is often ill-advised for three reasons. First, this approach has caused some authors to incorrectly anchor the camera frame to the system's physical focal plane, which places the system's center of perspective at the incorrect location for an external observer. Second, and without loss of generality, more intuitive results may be obtained by expressing the problem geometry in the image plane (instead of in the focal plane). Third, the parameters necessary to relate pixel coordinates and dimensioned focal plane coordinates are not observable in a system-level calibration. These ideas are now made explicit.

1) THE PINHOLE CAMERA MODEL's RELATION TO A PHYSICAL CAMERA SYSTEM

A review of the spacecraft OPNAV literature reveals some confusion about the geometry of image formation. This has led to widespread misrepresentation of how the pinhole camera model is related to a real-life camera system.

It is often convenient to represent an optical assembly consisting of many lenses by a single thin lens. This single thin lens, located at the optical assembly's rear principal plane and of the same size as the optical assembly's entrance pupil, provides a good approximation for many of the optical system's most important attributes. It makes sense, therefore, to consider the thin lens in more detail.

Consider a thin lens as shown in Fig. 2. The *optical center* of a single thin lens is the point along the optical axis where an incoming ray of light experiences no net deflection. This immediately gives rise to the pinhole camera model, with the usual argument going something like this [27], [28]: Suppose we have a thin lens that focuses light from an observed object onto the focal plane, where we observe the center ray experiences no deflection. Therefore, to model where an object will appear on the focal plane, imagine that the lens were to shrink to a point (or a *pinhole*) such that only this single undistorted ray passes through. This simplified geometry is the pinhole camera model.

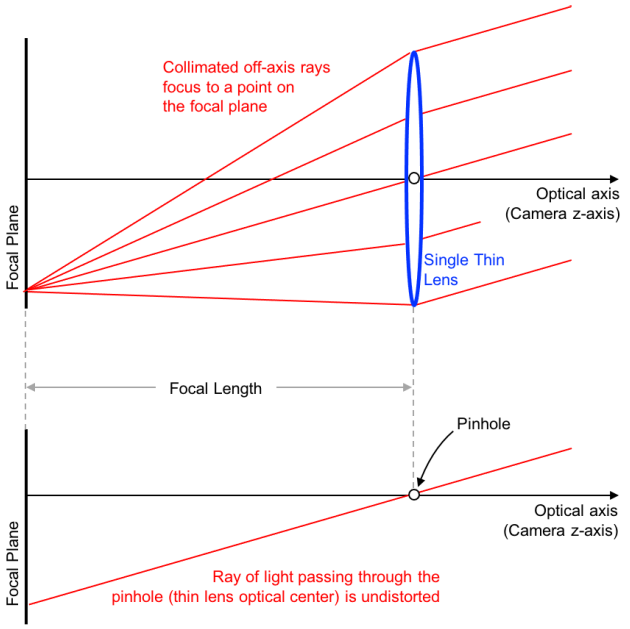

FIGURE 2. Illustration of the reduction of a single thin lens to the pinhole camera model.

With an optical assembly consisting of multiple lenses (as is always the case in practice), a true pinhole location does not exist. That is, there is no point in space through which arriving rays of light may pass and strike the correct location on the focal plane. The trouble is that the apparent pinhole location outside the camera (center of entrance pupil) is usually different from the apparent pinhole location inside the camera (rear nodal point). This causes the focal plane to appear to be at a different location than its true physical location when looking through the lens. The only way to recover the pinhole geometry is to break apart the camera and to pretend the focal plane lies at its apparent location, which is a distance of the effective focal length behind the center of the entrance pupil (see Fig. 3).

Recognizing that some of these optics concepts may be foreign to the usual spacecraft navigator, the situation for a real camera is made explicit by way of an example. Therefore, consider the example geometry shown in Fig. 3 for a double Gauss lens. The top four frames show the individual lenses outlined in dark blue, and the bottom two frames show how this collapses to a pinhole model. While the double Gauss lens is used in this example (it is a common compound lens configuration for cameras), we are not suggesting that a double Gauss lens is particularly good (or particularly bad) for any specific OPNAV purpose.

Most optical assemblies have an aperture stop that limits the amount of light that reaches the focal plane. If not a separate component (as illustrated in Fig. 3), there will always be some boundary that naturally acts as the aperture stop. An example chief ray (a meridional ray that passes through the center of the aperture stop) is illustrated as a solid red line in Figs. 3(a)–(d). The entrance pupil is the image (sometimes virtual) of the aperture stop as seen from outside the camera

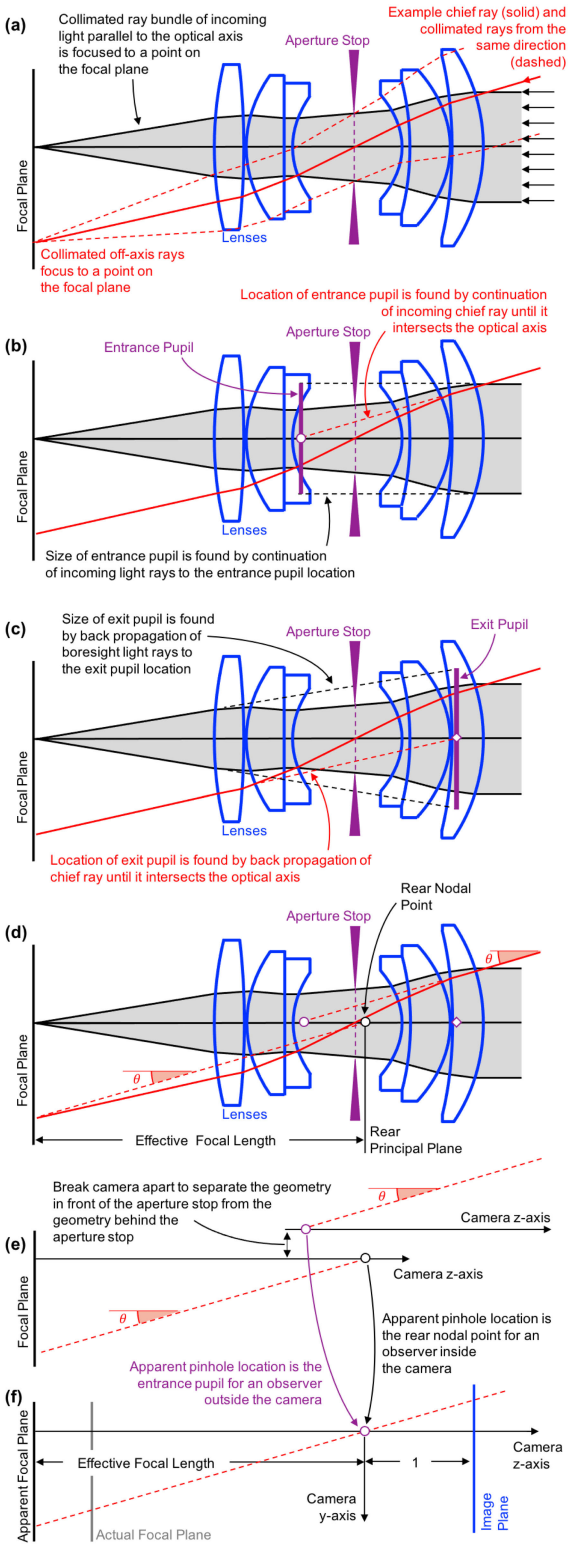

FIGURE 3. Illustration of the proper geometric reduction of an optical assembly to a pinhole camera model. Detailed discussion of each diagram is in the narrative. This particular example shows a double Gauss lens focused at infinity (optical design modeled after configuration from [29]).

looking inwards; see Fig. 3(b). Likewise, the exit pupil is the image (sometimes virtual) of the aperture stop as seen from inside the camera looking outwards; see Fig. 3(c).

It is clear from Fig. 3(b) and Fig. 3(c) that the camera's center of perspective is located at the entrance pupil for an observer outside the camera and at the exit pupil for an observer inside the camera. It is also clear that we can't directly use the exit pupil for constructing the pinhole camera model since the slope of the rays entering the optical assembly are usually different from the slope of the rays exiting the optical assembly; see Fig. 3(d). The difference in these slopes is a result of the system's pupil magnification, which may be written as the ratio of pupil diameters,

$$m_p = d_{xp}/d_{ep} \tag{1}$$

where d_{ep} is the entrance pupil diameter and d_{xp} is the exit pupil diameter. Since a camera's f-number N is defined as the ratio of focal length to entrance pupil diameter,

$$N = f/d_{ep} \tag{2}$$

we can write the distance s_{xp} between the exit pupil and the focal plane as

$$s_{xp} = m_p f = N d_{xp} \tag{3}$$

We note that the rear nodal point is coincident with the rear principal point in this example, where we've assumed the index of refraction is the same on both sides of the optical assembly.

Therefore, to construct a pinhole camera model, we must break the camera apart and accept a discontinuity across the pinhole location as shown in Fig. 3(e)–(f). We can then create what appears to be a pinhole camera by placing the origin at the center of the entrance pupil and pretending the focal plane lies a distance f behind this point. We find it useful in practice to abstract away the inside of the camera and work entirely in the image plane.

2) PARAMATERIZED CAMERA MODEL USING THE IMAGE PLANE

The image plane is a mathematical construct (not corresponding to any actual piece of sensor hardware) describing a planar surface that is parallel to the focal plane and that lies in front of the camera's center of projection. A variety of conventions exist, but the most popular (and most convenient) is to place the image plane at unit depth along the camera optical axis. Therefore, defining the camera frame's z-axis to be coincident with the optical axis (positive out of the camera), the focal plane is the $z = -f$ plane and the image plane is the $z = 1$ plane [see Fig. 3(f) and Fig. 4]. Following the conventions of [23], when looking out of the camera, we also choose the camera frame's x-axis to be positive to the right in the image and the y-axis to complete the right-handed system (down in the image).

The mathematics of perspective projection are best mechanized by operating in *projective space*, denoted here as $\mathbb{P}^n$. While the concept of a projective space is quite general [30], this tutorial only makes use of the real projective space—where $\mathbb{P}^n$ may be constructed as the quotient space of

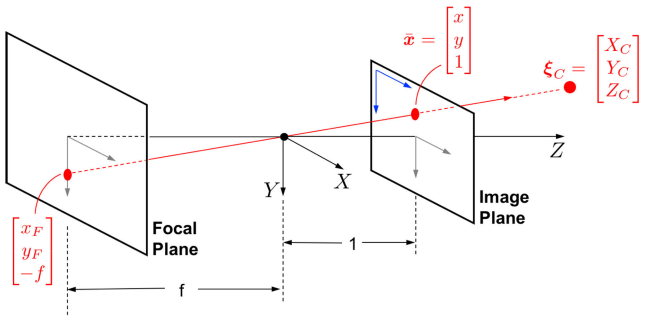

FIGURE 4. Illustration of 3D geometry for pinhole camera model, focal plane, and image plane. The origin of the camera frame is at the camera center. The x- and y-directions are redrawn on the focal plane and image plane for clarity. The ray connecting a 3D point at ξ_C and the camera center is shown in red, with the three key points along this ray from Eq. (4) shown by red dots.

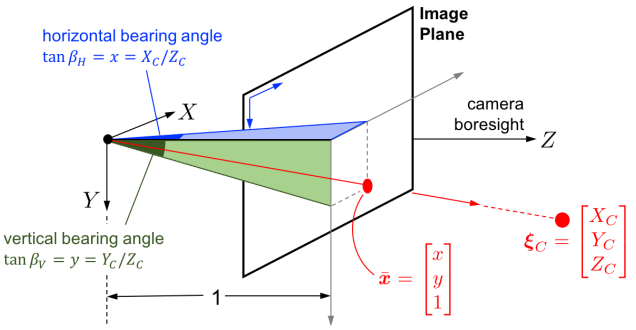

FIGURE 5. Illustration of image plane coordinates as tangent angles $x = \tan\beta_H$ and $y = \tan\beta_V$.

$\mathbb{R}^{n+1}\backslash\{0\}$ under the equivalence relation $\sim$ defined as

$$\boldsymbol{y} \sim k\boldsymbol{z}, \quad \{\boldsymbol{y}, \boldsymbol{z}\} \in \mathbb{R}^{n+1}\backslash\{0\}, \quad k \in \mathbb{R}\backslash\{0\}$$

Projective space is particularly important here because points in the image plane represent points in $\mathbb{P}^2$. Thus, in the special case of $\mathbb{P}^2$, we find that $n = 2$ and that $\{\boldsymbol{y}, \boldsymbol{z}\} \in \mathbb{R}^3$. Since the non-zero scalar k is arbitrary, we can gain some intuition about $\mathbb{P}^2$ by letting $\boldsymbol{z}$ be a unit vector (a direction) in $\mathbb{R}^3$. It follows that all the possible equivalent points $\boldsymbol{y}$ form a 3D line in the direction of $\boldsymbol{z}$ that passes through (but does not include) the origin. We will now explicitly connect these ideas to the pinhole camera model.

Suppose a camera observes a point in $\mathbb{R}^3$ with coordinates $\boldsymbol{\xi}_C^T = [X_C, Y_C, Z_C]$ as expressed in the camera frame. As can be seen in Fig. 4, the ray passing through both the camera's center of perspective (origin of camera frame) and the point $\boldsymbol{\xi}_C$ pierces the focal plane at coordinates $[x_F, y_F]$ and the image plane at coordinates $[x, y]$. Moreover, we know from the prior paragraph that every point along this ray (the red line in Fig. 4) represents the same point in $\mathbb{P}^2$. Consequently, since the three points in Fig. 4 belong to the same ray and differ by only a scalar multiple, it follows that

$$\begin{bmatrix} x_F \\ y_F \\ -f \end{bmatrix} \propto \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \propto \begin{bmatrix} X_C \\ Y_C \\ Z_C \end{bmatrix} \tag{4}$$

and that these three points in $\mathbb{R}^3$ correspond to the same point in $\mathbb{P}^2$.

The proportional relationships in Eq. (4) may be converted to equality relationships by taking ratios. The result is the familiar pinhole camera model expressions

$$x = -\frac{x_F}{f} = \frac{X_C}{Z_C} \quad \text{and} \quad y = -\frac{y_F}{f} = \frac{Y_C}{Z_C} \tag{5}$$

The middle terms in Eq. (5) are why the image plane coordinates $[x, y]$ are sometimes called *focal length normalized coordinates*. It is also sometimes convenient to interpret the image plane coordinates as the tangent of a pair of bearing angles

$$x = \tan\beta_H \quad \text{and} \quad y = \tan\beta_V \tag{6}$$

which is illustrated in Fig. 5.

When given a 2D point $[x, y]$ in the image plane, we may easily form the corresponding point in $\mathbb{P}^2$ by using homogeneous coordinates. Specifically, if $\boldsymbol{x}^T = [x, y]$ is a point's 2D image plane coordinate, then it may be written in homogeneous coordinates by appending a 1 to $\boldsymbol{x}$ to form the 3×1 vector $\bar{\boldsymbol{x}}^T = [x, y, 1]$. This works out nicely since we have conveniently chosen to place the image plane at unit depth (the $z = 1$ plane) in front of the camera center. Therefore, using Eq. (4), the pinhole camera model written in the camera frame is simply

$$\bar{\boldsymbol{x}} \propto \boldsymbol{\xi}_C \tag{7}$$

The general situation is somewhat more complicated since the observed 3D points are often known in a frame other than the camera frame. Suppose, for example, that we have a 3D point on the celestial body's surface at location $\boldsymbol{p}$ relative to the body's center. If we define $\boldsymbol{r}$ as the vector from the camera to the celestial body's center, then

$$\boldsymbol{\xi} = \boldsymbol{r} + \boldsymbol{p} \tag{8}$$

This relation, illustrated in Fig. 6, is true in any consistent frame. Therefore keeping $\boldsymbol{\xi}$ in the camera frame, we may consider the situation where $\boldsymbol{r}$ and $\boldsymbol{p}$ are known in a different frame (e.g., a planet-fixed frame P),

$$\boldsymbol{\xi}_C = \boldsymbol{r}_C + \boldsymbol{p}_C = \boldsymbol{T}_C^P \left(\boldsymbol{r}_P + \boldsymbol{p}_P\right) \tag{9}$$

where $\boldsymbol{T}_C^P$ is the attitude transformation matrix (passive interpretation of a rotation [31]) that transforms a vector expressed in frame P to that same vector expressed in frame C. Defining the projection matrix $\boldsymbol{P}_C^P$ as

$$\boldsymbol{P}_C^P = \boldsymbol{T}_C^P \begin{bmatrix} \boldsymbol{I}_{3\times3} & \boldsymbol{r}_P \end{bmatrix} = \begin{bmatrix} \boldsymbol{T}_C^P & \boldsymbol{r}_C \end{bmatrix} \tag{10}$$

one may write $\boldsymbol{p}_P$ in homogeneous coordinates (i.e., written as a point in $\mathbb{P}^3$)

$$\bar{\boldsymbol{p}}_P = \begin{bmatrix} \boldsymbol{p}_P \\ 1 \end{bmatrix} \tag{11}$$

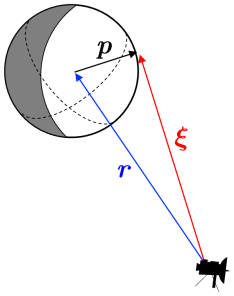

FIGURE 6. Illustration of key vectors for a spacecraft observing a point on the surface of a celestial body.

to finally obtain

$$\boldsymbol{\xi}_C = \boldsymbol{P}_C^P \bar{\boldsymbol{p}}_P \tag{12}$$

Substituting this result into the pinhole camera model from Eq. (7) we find that

$$\bar{\boldsymbol{x}} \propto \boldsymbol{\xi}_C = \boldsymbol{P}_C^P \bar{\boldsymbol{p}}_P \tag{13}$$

Thus, in general, we see the pinhole camera model describes a mapping from $\mathbb{P}^3$ to $\mathbb{P}^2$.

3) RELATING IMAGE PLANE AND PIXEL COORDINATES

To make use of a digital image, we must be able to relate points in an image plane to their corresponding point in a digital image. This is straightforward to do with a simple affine transformation. While the discussion that follows includes some discussion of geometry on the focal plane, this is for derivation and context purposes only—the reader is encouraged to work entirely in the image plane.

We begin by defining the u-v coordinate system in the digital image. When looking at an image (out of the camera and in the direction of the image plane; e.g., what you might expect to see looking at an image on your computer screen), we place the origin at the center of the upper-left pixel with the u-direction to the right and the v-direction down. Thus, by definition, integer values of $[u, v]$ exist at pixel centers with u corresponding to column number and v corresponding to row number. This is shown in the right-hand frame of Fig. 7.

As mentioned briefly before, we choose the camera frame z-axis to be coincident with the camera optical axis and positive pointing out of the camera. Now, without loss of generality, let the camera frame x-axis point in the same direction as u (to the right in the image). The y-axis completes the right-hand system (down in the image).

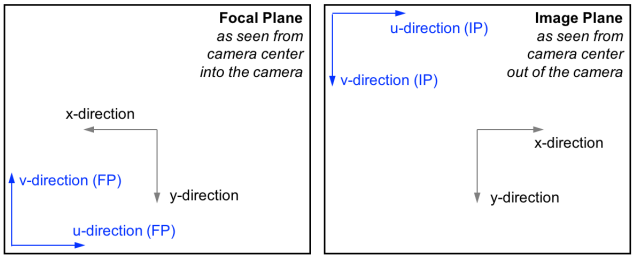

FIGURE 7. Illustration of focal plane and image plane.

To construct the required conversion, first consider the geometry on the focal plane where the FPA sensor resides. Define the pixel coordinates of the focal plane's principal point (where the optical axis pierces the focal plane) to be $[u_p, v_p]$.

Let μ_x and μ_y be the distance from pixel center to pixel center (sometimes called the *pixel pitch*) in the x and y directions. Using the coordinate frame arrangements from the left-hand frame of Fig. 7, the conversion from focal plane coordinates $[x_F, y_F]$ to pixel coordinates $[u, v]$ is simply

$$u = u_p - \frac{x_F}{\mu_x} \quad \text{and} \quad v = v_p - \frac{y_F}{\mu_y} \tag{14}$$

This can be converted from focal plane coordinates to image plane coordinates by substitution from Eq. (5),

$$u = \frac{f}{\mu_x} x + u_p \quad \text{and} \quad v = \frac{f}{\mu_y} y + v_p \tag{15}$$

Defining $d_x = f/\mu_x > 0$ and $d_y = f/\mu_y > 0$,

$$u = d_x x + u_p \quad \text{and} \quad v = d_y y + v_p \tag{16}$$

which we observe to be consistent with the image plane coordinate frame conventions in the right-hand frame of Fig. 7. Allowing for some affine shear and keeping the x-direction and u-direction parallel to one another,

$$u = d_x x + \alpha y + u_p \quad \text{and} \quad v = d_y y + v_p \tag{17}$$

This may be compactly rewritten in homogeneous coordinates as

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} d_x & \alpha & u_p \\ 0 & d_y & v_p \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{18}$$

or, more compactly,

$$\bar{\boldsymbol{u}} = \boldsymbol{K}\bar{\boldsymbol{x}} \tag{19}$$

where $\bar{\boldsymbol{u}}^T = [u, v, 1]$. The 3×3 matrix $\boldsymbol{K}$ is called the *camera calibration matrix* and describes the affine transformation from image plane coordinates $[x, y]$ to pixel coordinates $[u, v]$. Since $\boldsymbol{K}$ is full rank, it may be inverted to find

$$\bar{\boldsymbol{x}} = \boldsymbol{K}^{-1}\bar{\boldsymbol{u}} \tag{20}$$

Moreover, the structure of $\boldsymbol{K}$ permits simple computation of the inverse as [5]

$$\boldsymbol{K}^{-1} = \begin{bmatrix} \frac{1}{d_x} & \frac{-\alpha}{d_x d_y} & \frac{\alpha v_p - d_y u_p}{d_x d_y} \\ 0 & \frac{1}{d_y} & \frac{-v_p}{d_y} \\ 0 & 0 & 1 \end{bmatrix} \tag{21}$$

Using Eq. (19) and Eq. (20), it is possible to work entirely in the image plane, without ever needing to explicitly consider the focal plane. Moreover, this makes clear that only

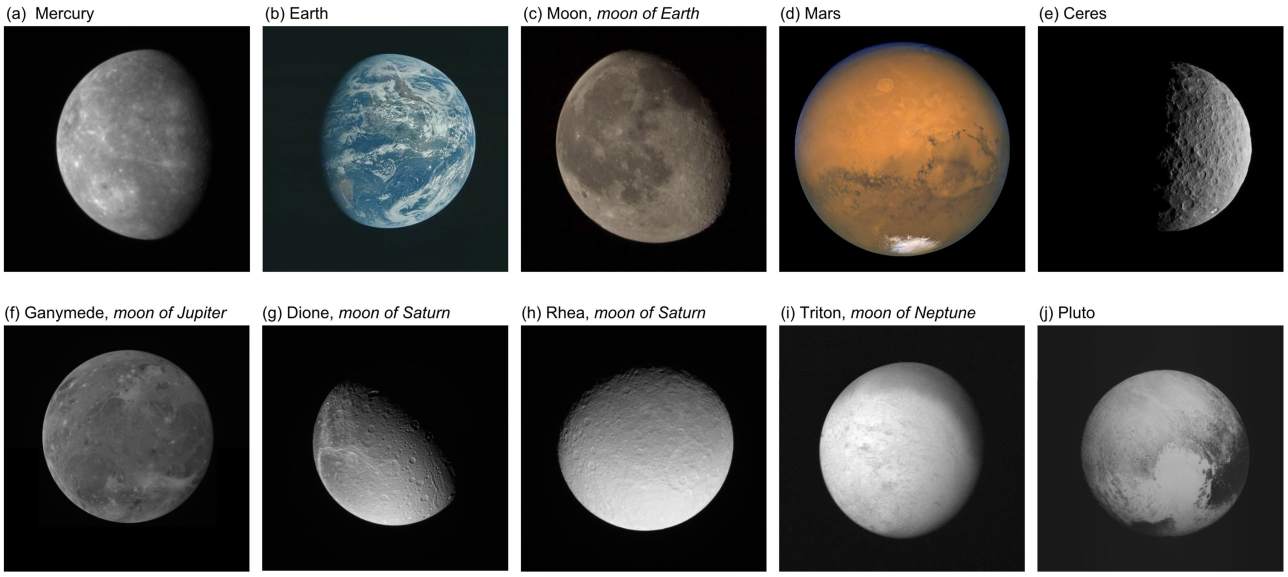

FIGURE 8. Examples of celestial bodies that are well-modeled as a sphere, spheroid, or triaxial ellipsoid at the global level. All examples shown here are real images collected by past exploration spacecraft: (a) Mercury, image CN0108893564M_RA_5 from MESSENGER spacecraft [34]; (b) Earth, Apollo 15 image AS15-91-12343 [credit: NASA]; (c) Moon, Space Shuttle image STS128-E-9477 [credit: NASA]; (d) Mars, image from Hubble Space Telescope [credit: NASA/J. Bell/M. Wolff]; (e) Ceres, image FC21A0034697 from Dawn spacecraft [35]; (f) Ganymede, image 3078r from Galileo spacecraft [36]; (g) Dione, image N1626025735 from Cassini spacecraft [37]; (h) Rhea, image N1598020836 from Cassini spacecraft [37]; (i) Triton, image C1137709_CALIB from Voyager 2 spacecraft [38], [39]; (j) Pluto, image lor_0299127143_0x630_eng from New Horizons spacecraft [40]. Original images have been cropped.

the ratios $d_x = f/\mu_x$ and $d_y = f/\mu_y$ are important. Indeed, in a typical camera calibration (e.g., Refs. [23], [32], [33]), it is impossible to obtain numerical values for the focal length or the pixel pitch without assuming the other is known.

For any real imaging system, values for the camera calibration matrix $\boldsymbol{K}$ should be computed from a camera calibration procedure. Simply computing them from vendor specifications is rarely accurate enough for practical OPNAV. The reader interested in geometric camera calibration for a space imaging system is directed to [23].

B. GLOBAL SHAPE OF LARGE CELESTIAL BODIES

The global shapes of the planets, dwarf planets, and many of their moons are well-modeled as a sphere, spheroid, or triaxial ellipsoid [41]. This fact may be qualitatively observed by simply looking at real images of large celestial bodies, such as those shown in Fig. 8. The natural tendency of large bodies to take a rounded shape may also be shown quantitatively by consideration of gravitational potential and rotational dynamics [42]. This has been known for some time—indeed, it was shown by Newton in the *Principia* [43] that the global shape of a self-gravitating and rotating body in hydrostatic equilibrium is an oblate spheroid.

There are a variety of ways to mathematically represent an ellipsoidal surface (which, in our case, describes a celestial body's shape). This work chooses to describe ellipsoids using the idea of a *quadric surface*.

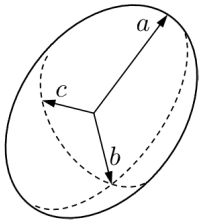

FIGURE 9. Illustration of quadric locus for a triaxial ellipsoid with principal axis dimensions $a > b > c$.

1) CELESTIAL BODIES AS A QUADRIC LOCUS

Consider an ellipsoidal celestial body with principal axis dimensions of a, b, and c, as shown in Fig. 9. A point $[X, Y, Z]$ on the celestial body's surface in the body's principal axis frame is simply

$$\frac{X^2}{a^2} + \frac{Y^2}{b^2} + \frac{Z^2}{c^2} = 1 \tag{22}$$

Without loss of generality, we usually define the principal axis frame such that $a \geq b \geq c$. A body is a sphere when $a = b = c$, an oblate spheroid (i.e., an ellipse of revolution that is spun around its minor axis) when $a = b > c$, and a triaxial ellipsoid when $a > b > c$.

Therefore, denote the vector from the center of the celestial body to its surface by the 3×1 vector $\boldsymbol{p}^T = [X, Y, Z]$, such that in the principal axis frame we may write

$$\boldsymbol{p}_P^T \boldsymbol{A}_P \boldsymbol{p}_P = 1 \tag{23}$$

where $\boldsymbol{A}_P$ is a 3×3 diagonal matrix of full rank

$$\boldsymbol{A}_P = \begin{bmatrix} 1/a^2 & 0 & 0 \\ 0 & 1/b^2 & 0 \\ 0 & 0 & 1/c^2 \end{bmatrix} \tag{24}$$

and where the subscript P denotes the principal axis frame. Using the convention $a \geq b \geq c$ produces a diagonal of $\boldsymbol{A}_P$ that is sorted in ascending order and a diagonal of $\boldsymbol{A}_P^{-1}$ that is sorted in descending order.

We may express the vector $\boldsymbol{p}$ in a frame aligned with the camera's coordinate frame

$$\boldsymbol{p}_C = \boldsymbol{T}_C^P \boldsymbol{p}_P \tag{25}$$

where, as before, $\boldsymbol{T}_C^P$ is the attitude transformation matrix (passive interpretation of a rotation [31]) that transforms a vector expressed in frame P to that same vector expressed in frame C. Equivalently, we also have

$$\boldsymbol{p}_P = \boldsymbol{T}_P^C \boldsymbol{p}_C \tag{26}$$

Substitution of Eq. (26) into Eq. (23) yields

$$\boldsymbol{p}_C^T \boldsymbol{T}_C^P \boldsymbol{A}_P \boldsymbol{T}_P^C \boldsymbol{p}_C = \boldsymbol{p}_C^T \boldsymbol{A}_C \boldsymbol{p}_C = 1 \tag{27}$$

Consequently, the shape of the ellipsoidal celestial body (relative to its center) in a frame aligned with the camera frame is given by the symmetric 3×3 matrix $\boldsymbol{A}_C$

$$\boldsymbol{A}_C = \boldsymbol{T}_C^P \boldsymbol{A}_P \boldsymbol{T}_P^C \tag{28}$$

This article is about cameras imaging an ellipsoidal celestial body. Therefore, rather than working in $\mathbb{R}^3$, we find it useful to describe the celestial body as a quadric surface in $\mathbb{P}^3$. This allows us to more easily exploit some elegant results from algebraic projective geometry [44]. Thus, moving to projective space, write the planet-centered surface point $\boldsymbol{p}$ in homogeneous coordinates, $\bar{\boldsymbol{p}}^T = [\boldsymbol{p}^T \ 1] \in \mathbb{P}^3$.

We can now rewrite Eq. (23) in homogeneous coordinates

$$\bar{\boldsymbol{p}}_P^T \begin{bmatrix} \boldsymbol{A}_P & 0_{3\times 1} \\ 0_{1\times 3} & -1 \end{bmatrix} \bar{\boldsymbol{p}}_P \equiv \bar{\boldsymbol{p}}_P^T \boldsymbol{Q}_P \bar{\boldsymbol{p}}_P = 0 \tag{29}$$

where $\boldsymbol{Q}_P$ is a 4×4 diagonal matrix of full rank that describes the surface of the celestial body as a quadric locus. When $\boldsymbol{Q}$ is full rank we call the set of points lying on this surface a *proper quadric locus*.

Rather than writing the quadric surface in the celestial body-centered principal axis frame P, we would like to describe the quadric as it appears to an observer at the origin of the camera frame C. This requires both an origin shift and a rotation, which is easily handled by a single operation when working in homogeneous coordinates. Therefore, recalling the geometry of Fig. 6 and rearranging Eq. (9), we have

$$\boldsymbol{p}_P = \boldsymbol{T}_P^C \left(\boldsymbol{\xi}_C - \boldsymbol{r}_C \right) = \boldsymbol{T}_P^C \boldsymbol{\xi}_C - \boldsymbol{r}_P \tag{30}$$

In homogeneous coordinates this is

$$\bar{\boldsymbol{p}}_P = \begin{bmatrix} \boldsymbol{T}_P^C & -\boldsymbol{r}_P \\ 0_{1\times 3} & 1 \end{bmatrix} \bar{\boldsymbol{\xi}}_C \tag{31}$$

Substituting this result for $\bar{\boldsymbol{p}}_P$ in Eq. (29) leads to

$$\bar{\boldsymbol{\xi}}_C^T \begin{bmatrix} \boldsymbol{T}_C^P \boldsymbol{A}_P \boldsymbol{T}_P^C & -\boldsymbol{T}_C^P \boldsymbol{A}_P \boldsymbol{r}_P \\ -\boldsymbol{r}_P^T \boldsymbol{A}_P^T \boldsymbol{T}_P^C & \boldsymbol{r}_P^T \boldsymbol{A}_P \boldsymbol{r}_P - 1 \end{bmatrix} \bar{\boldsymbol{\xi}}_C = \bar{\boldsymbol{\xi}}_C^T \begin{bmatrix} \boldsymbol{A}_C & -\boldsymbol{A}_C \boldsymbol{r}_C \\ -\boldsymbol{r}_C^T \boldsymbol{A}_C^T & \boldsymbol{r}_C^T \boldsymbol{A}_C \boldsymbol{r}_C - 1 \end{bmatrix} \bar{\boldsymbol{\xi}}_C = 0 \tag{32}$$

We acknowledge that $\boldsymbol{A}_C = \boldsymbol{A}_C^T$, but we keep the transpose in the lower left term to help emphasize the problem's symmetry. It follows that the celestial body's quadric locus as seen by the camera is given by the 4×4 symmetric matrix of full rank $\boldsymbol{Q}_C$,

$$\boldsymbol{Q}_C = \begin{bmatrix} \boldsymbol{A}_C & -\boldsymbol{A}_C \boldsymbol{r}_C \\ -\boldsymbol{r}_C^T \boldsymbol{A}_C^T & \boldsymbol{r}_C^T \boldsymbol{A}_C \boldsymbol{r}_C - 1 \end{bmatrix} \tag{33}$$

such that one may write

$$\bar{\boldsymbol{\xi}}_C^T \boldsymbol{Q}_C \bar{\boldsymbol{\xi}}_C = 0 \tag{34}$$

2) CELESTIAL BODIES AS A QUADRIC ENVELOPE

If a proper quadric locus is the set of all points $\bar{\boldsymbol{\xi}}$ lying on an ellipsoidal body's surface, let the proper *quadric envelope* be the set of all planes $\boldsymbol{\pi}$ that are tangent to the ellipsoidal body's surface. As we will see, it is sometimes easier to deal with the quadric envelope than the quadric locus in derivations involving projective geometry.

Given a proper quadric locus $\boldsymbol{Q}$, the corresponding quadric envelope is described by the adjugate matrix $\boldsymbol{Q}^*$. To see this, define a plane by the 4×1 vector $\boldsymbol{\pi}$. A 3D point written in homogeneous coordinates $\bar{\boldsymbol{\xi}}$ lies on the plane when

$$\boldsymbol{\pi}^T \bar{\boldsymbol{\xi}} = 0 \tag{35}$$

The reader may quickly verify to themself that the plane $\boldsymbol{\pi} \propto \boldsymbol{Q}\bar{\boldsymbol{\xi}}$ is tangent to the quadric $\boldsymbol{Q}$ at point $\bar{\boldsymbol{\xi}}$.

Applying these ideas to the quadric $\boldsymbol{Q}_C$ and observing the matrix to be full rank,

$$\bar{\boldsymbol{\xi}}_C \propto \boldsymbol{Q}_C^{-1} \boldsymbol{\pi}_C \tag{36}$$

Substituting this result into Eq. (34),

$$\bar{\boldsymbol{\xi}}_C^T \boldsymbol{Q}_C \bar{\boldsymbol{\xi}}_C = \boldsymbol{\pi}_C^T \boldsymbol{Q}_C^{-T} \boldsymbol{Q}_C \boldsymbol{Q}_C^{-1} \boldsymbol{\pi}_C = 0 \tag{37}$$

which simplifies to

$$\boldsymbol{\pi}_C^T \boldsymbol{Q}_C^{-1} \boldsymbol{\pi}_C = 0 \tag{38}$$

Using Eq. (33), we may analytically compute $\boldsymbol{Q}_C^{-1}$ to be

$$\boldsymbol{Q}_C^{-1} = \begin{bmatrix} \boldsymbol{A}_C^{-1} - \boldsymbol{r}_C \boldsymbol{r}_C^T & -\boldsymbol{r}_C \\ -\boldsymbol{r}_C^T & -1 \end{bmatrix} \tag{39}$$

To proceed, recall that the scaling of $\boldsymbol{Q}_C$ and $\boldsymbol{Q}_C^*$ is arbitrary. Moreover, recall that the adjugate $\boldsymbol{Q}_C^*$ is related to the inverse by

$$\boldsymbol{Q}_C^* = \det(\boldsymbol{Q}_C) \boldsymbol{Q}_C^{-1} \propto \boldsymbol{Q}_C^{-1} \tag{40}$$

where we have relied on the knowledge that the determinant of $\boldsymbol{Q}_C$ is never nearly zero for a real celestial body

$$\det(\boldsymbol{Q}_C) = \det(\boldsymbol{Q}_P) = -\det(\boldsymbol{A}_P) = -(abc)^{-2} \not\approx 0 \tag{41}$$

Substituting Eq. (40) into Eq. (38) yields the usual expression for a quadric envelope

$$\boldsymbol{\pi}_C^T \boldsymbol{Q}_C^* \boldsymbol{\pi}_C = 0 \tag{42}$$

The reader will observe Eqs. (34) and (42) to be one example of the duality between points and planes in $\mathbb{P}^3$.

C. CELESTIAL BODY'S TIGHTLY BOUNDING CONE AND THE APPARENT HORIZON

Every point in an image may be thought of as a point in $\mathbb{P}^2$. Thus, every image point represents a ray passing through the origin of the camera frame. If we take the collection of rays emanating from the camera that are tangent to the celestial body, we get the celestial body's *tightly bounding cone* (see Fig. 10). The tightly bounding cone is an example of a proper quadric cone which is described by a 4×4 symmetric matrix of rank three [44].

An ellipsoid's tightly bounding cone with apex at homogeneous coordinate $\bar{\boldsymbol{a}}$ is defined by the quadric cone $\boldsymbol{X}$ [45]

$$\boldsymbol{X} \propto (\boldsymbol{Q}\bar{\boldsymbol{a}})\,(\boldsymbol{Q}\bar{\boldsymbol{a}})^T - \left(\bar{\boldsymbol{a}}^T \boldsymbol{Q}\bar{\boldsymbol{a}}\right)\boldsymbol{Q} \tag{43}$$

Any point $\bar{\boldsymbol{\xi}} \in \mathbb{P}^3$ that satisfies the constraint $\bar{\boldsymbol{\xi}}^T \boldsymbol{X} \bar{\boldsymbol{\xi}} = 0$ lies on this cone. This constraint and the expression in Eq. (43) is valid in any consistent frame.

Proceed by considering the geometry in the camera frame. Therefore, we have the proper quadric locus $\boldsymbol{Q}_C$ from Eq. (33) and we choose to place the cone apex at the camera. Since the camera is at the origin in frame C, the apex coordinate is simply $\bar{\boldsymbol{a}}_C^T = [0, 0, 0, 1]$ and

$$\boldsymbol{Q}_C \bar{\boldsymbol{a}}_C = \begin{bmatrix} -\boldsymbol{A}_C \boldsymbol{r}_C \\ \boldsymbol{r}_C^T \boldsymbol{A}_C \boldsymbol{r}_C - 1 \end{bmatrix} \tag{44}$$

$$\bar{\boldsymbol{a}}_C^T \boldsymbol{Q}_C \bar{\boldsymbol{a}}_C = \boldsymbol{r}_C^T \boldsymbol{A}_C \boldsymbol{r}_C - 1 \tag{45}$$

which directly leads to

$$\boldsymbol{X}_C \propto \begin{bmatrix} \boldsymbol{M}_C & 0_{3\times1} \\ 0_{1\times3} & 0 \end{bmatrix} \tag{46}$$

where

$$\boldsymbol{M}_C = \boldsymbol{A}_C \boldsymbol{r}_C \boldsymbol{r}_C^T \boldsymbol{A}_C - (\boldsymbol{r}_C^T \boldsymbol{A}_C \boldsymbol{r}_C - 1)\boldsymbol{A}_C \tag{47}$$

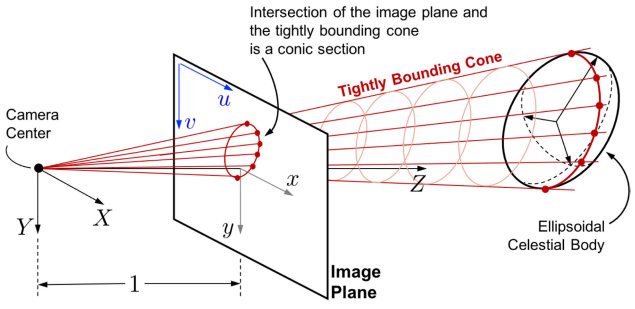

FIGURE 10. Illustration of a celestial body's tightly bounding cone and the conic section (apparent horizon) formed by the intersection of this cone with the image plane.

The equation for $\boldsymbol{M}_C$ can also be derived without the use of quadrics and homogeneous coordinates, as was done in [46].

The rays belonging to the tightly bounding cone are the rays emanating from the camera location that are tangent to the celestial body. Thus, they are the rays that form the apparent horizon. The intersection of this cone with the image plane forms a conic section. This is also illustrated in Fig. 10.

D. PARAMETERIZATION OF A CONIC (HORIZON) ARC IN AN IMAGE

The apparent horizon of an ellipsoidal celestial body in an image forms a conic section. Thus, the shape of the horizon arc must be a circle, ellipse, parabola, or hyperbola. In this section we will discuss the appropriate parameterization of a conic section for the image-based navigation problem. Later (in Section II-E3) we discuss the constraints on when the different horizon shapes are practically realizable.

A generic conic section is given by the implicit quadratic equation

$$Ax^2 + Bxy + Cy^2 + Dx + Fy + G = 0 \tag{48}$$

which describes a closed conic (i.e., a circle or an ellipse) when $B^2 - 4AC < 0$ and a circle when $(A - C)^2 + B^2 = 0$.

1) CONICS IN THE IMAGE PLANE

Since the image plane is conveniently placed at unit depth and $\bar{\boldsymbol{x}}^T = [x, y, 1]$, we may write Eq. (48) as

$$\bar{\boldsymbol{x}}^T \boldsymbol{C}\, \bar{\boldsymbol{x}} = 0 \tag{49}$$

where the symmetric 3×3 matrix $\boldsymbol{C}$ is given by

$$\boldsymbol{C} = \begin{bmatrix} A & B/2 & D/2 \\ B/2 & C & F/2 \\ D/2 & F/2 & G \end{bmatrix} \tag{50}$$

The set of points $\bar{\boldsymbol{x}}$ satisfying Eq. (49) define the *conic locus*, as illustrated in Fig. 11.

Define a line by the 3×1 vector $\boldsymbol{\ell}$, such that a point $\bar{\boldsymbol{x}}$ lies on the line if $\boldsymbol{\ell}^T \bar{\boldsymbol{x}} = 0$. Observe that the line $\boldsymbol{\ell} \propto \boldsymbol{C}\bar{\boldsymbol{x}}$ is tangent to the conic at point $\bar{\boldsymbol{x}}$. Since $\boldsymbol{C}$ is full rank, we find that $\bar{\boldsymbol{x}} \propto \boldsymbol{C}^{-1}\boldsymbol{\ell}$. Therefore.

$$\boldsymbol{\ell}^T \boldsymbol{C}^{-T} \boldsymbol{C} \boldsymbol{C}^{-1} \boldsymbol{\ell} = \boldsymbol{\ell}^T \boldsymbol{C}^{-1} \boldsymbol{\ell} = 0 \tag{51}$$

As with the discussion of the quadric envelope, observe that $\det(\boldsymbol{C}) \not\approx 0$, such that Eq. (51) is the same as

$$\boldsymbol{\ell}^T \boldsymbol{C}^* \boldsymbol{\ell} = 0 \tag{52}$$

where $\boldsymbol{C}^*$ is the adjugate of $\boldsymbol{C}$. The set of lines $\boldsymbol{\ell}$ satisfying Eq. (52) define the *conic envelope*, which is also illustrated in Fig. 11. The reader will observe Eqs. (49) and (52) to be one example of the duality between points and lines in $\mathbb{P}^2$.

2) RELATION OF IMAGE PLANE CONICS TO PIXEL SPACE CONICS

This tutorial primarily works with conics expressed in image plane coordinates. We denote these conics using $\boldsymbol{C}$ [for a

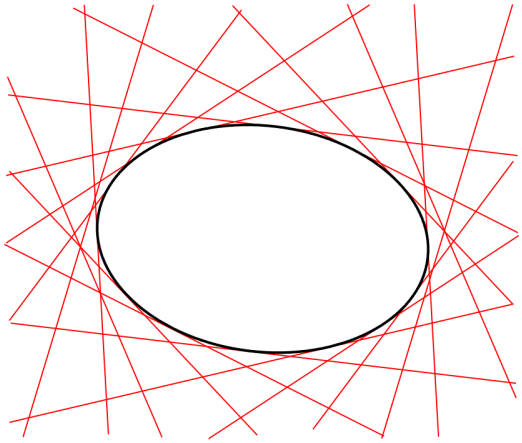

FIGURE 11. Illustration of a conic locus (black ellipse) and a sampling of lines (red) belonging to the conic envelope.

conic locus, see Eq. (49)] and $\boldsymbol{C}^*$ [for a conic envelope, see Eq. (52)]. This convention is used without loss of generality.

It is sometimes necessary to relate the conic in image plane coordinates to the same conic in pixel coordinates. Since we assume a calibrated camera, the camera calibration matrix $\boldsymbol{K}$ is known—thus we may freely move between an image plane coordinate $\bar{\boldsymbol{x}}$ and a pixel coordinate $\bar{\boldsymbol{u}}$ using Eq. (19) and Eq. (20).

To find the expression to transformation a conic between these different coordinate systems, substitute Eq. (20) into Eq. (49),

$$\bar{\boldsymbol{x}}^T\boldsymbol{C}\,\bar{\boldsymbol{x}} = \bar{\boldsymbol{u}}^T\boldsymbol{K}^{-T}\boldsymbol{C}\boldsymbol{K}^{-1}\bar{\boldsymbol{u}} = \bar{\boldsymbol{u}}^T\boldsymbol{C}_{uv}\,\bar{\boldsymbol{u}} = 0 \tag{53}$$

where $\boldsymbol{C}_{uv}$ is a 3×3 matrix in the form of Eq. (50) that describes the conic locus in pixel coordinates. Thus, we may convert between $\boldsymbol{C}$ and $\boldsymbol{C}_{uv}$ using

$$\boldsymbol{K}^{-T}\boldsymbol{C}\boldsymbol{K}^{-1} \propto \boldsymbol{C}_{uv} \quad \longleftrightarrow \quad \boldsymbol{C} \propto \boldsymbol{K}^T\boldsymbol{C}_{uv}\boldsymbol{K} \tag{54}$$

Inverting these expressions allows us to develop equivalent relations for the conic envelope

$$\boldsymbol{K}\boldsymbol{C}^*\boldsymbol{K}^T \propto \boldsymbol{C}^*_{uv} \quad \longleftrightarrow \quad \boldsymbol{C}^* \propto \boldsymbol{K}^{-1}\boldsymbol{C}^*_{uv}\boldsymbol{K}^{-T} \tag{55}$$

Consequently we may freely move between a conic in image plane coordinates (described by $\boldsymbol{C}$ and $\boldsymbol{C}^*$) and the same conic in pixel coordinates (described by $\boldsymbol{C}_{uv}$ and $\boldsymbol{C}^*_{uv}$).

E. COMPUTING HORIZON PROJECTIONS

Computing an ellipsoidal body's horizon arc in an image is fundamentally a conic section problem. We now discuss how to analytically compute the apparent horizon using both the quadric/conic locus and the quadric/conic envelope. This naturally leads to physical constraints on the types of horizon arcs visible in any situation.

1) COMPUTING THE IMAGE HORIZON WITH THE QUADRIC LOCUS

A celestial body's tightly bounding cone may be formed by the back projection of the horizon's image ellipse. The back projection of an image conic is given by [45]

$$\boldsymbol{X}_{(\cdot)} \propto \left[\boldsymbol{P}_C^{(\cdot)}\right]^T \boldsymbol{C}\boldsymbol{P}_C^{(\cdot)} \tag{56}$$

If we choose to operate in the camera frame, then the projection matrix of interest [see Eq. (10)] is simply

$$\boldsymbol{P}_C^C = \begin{bmatrix}\boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times1}\end{bmatrix} \tag{57}$$

Thus, combining this results with the expression for $\boldsymbol{X}_C$ from Eq. (46), we find that

$$\boldsymbol{X}_C \propto \begin{bmatrix}\boldsymbol{M}_C & \boldsymbol{0}_{3\times1}\\ \boldsymbol{0}_{1\times3} & 0\end{bmatrix} \propto \begin{bmatrix}\boldsymbol{C} & \boldsymbol{0}_{3\times1}\\ \boldsymbol{0}_{1\times3} & 0\end{bmatrix} \tag{58}$$

After substituting $\boldsymbol{M}_C$ from Eq. (47), this produces an analytic expression for the horizon's conic locus in an image

$$\boldsymbol{C} \propto \boldsymbol{M}_C = \boldsymbol{A}_C\boldsymbol{r}_C\boldsymbol{r}_C^T\boldsymbol{A}_C - (\boldsymbol{r}_C^T\boldsymbol{A}_C\boldsymbol{r}_C - 1)\boldsymbol{A}_C \tag{59}$$

2) COMPUTING THE IMAGE HORIZON WITH THE QUADRIC ENVELOPE

The action of a projective camera on a quadric is most easily written in terms of the quadric and conic envelopes [45],

$$\boldsymbol{C}^* \propto \boldsymbol{P}_C^{(\cdot)}\boldsymbol{Q}^*_{(\cdot)}\left[\boldsymbol{P}_C^{(\cdot)}\right]^T \tag{60}$$

Given $\boldsymbol{Q}^*_C \propto \boldsymbol{Q}_C^{-1}$ from Eq. (39) and $\boldsymbol{P}_C^C$ from Eq. (57), we find that

$$\boldsymbol{C}^* \propto \boldsymbol{A}_C^{-1} - \boldsymbol{r}_C\boldsymbol{r}_C^T \tag{61}$$

Application of the Sherman–Morrison formula produces the expected result

$$\boldsymbol{C}^* \propto \boldsymbol{A}_C^{-1} - \boldsymbol{r}_C\boldsymbol{r}_C^T = (1 - \boldsymbol{r}_C^T\boldsymbol{A}_C\boldsymbol{r}_C)\boldsymbol{M}_C^{-1} \propto \boldsymbol{M}_C^{-1} \tag{62}$$

3) CONSTRAINTS ON HORIZON CONIC TYPES

In most cases, the apparent horizon arc for ellipsoidal bodies is an ellipse at large distances (e.g., OPNAV when the entire celestial body appears in the image) and is a hyperbola at small distances (e.g., horizon-based attitude determination in LEO). The specific conditions required for various horizon shapes are now discussed.

At first, it may not be apparent how the image of a finite body can result in an open horizon arc (e.g., a hyperbola). We show this here by considering the simple situation of imaging a nearly spherical celestial body. For a spherical body we know the principal axis lengths to be $a = b = c$ such that $\boldsymbol{A}_C = \boldsymbol{A}_P = (1/a^2)\boldsymbol{I}_3$. Therefore, from Eq. (59), we find that

$$\boldsymbol{C} \propto \frac{1}{a^4}\boldsymbol{r}_C\boldsymbol{r}_C^T - \frac{1}{a^4}(\boldsymbol{r}_C^T\boldsymbol{r}_C - a^2)\boldsymbol{I}_3 \tag{63}$$

Letting $\boldsymbol{r}_C^T = [X_C, Y_C, Z_C]$, this becomes

$$\boldsymbol{C} \propto \begin{bmatrix} a^2 - Y_C^2 - Z_C^2 & X_CY_C & X_CZ_C \\ X_CY_C & a^2 - X_C^2 - Z_C^2 & Y_CZ_C \\ X_CZ_C & Y_CZ_C & a^2 - X_C^2 - Y_C^2 \end{bmatrix} \tag{64}$$

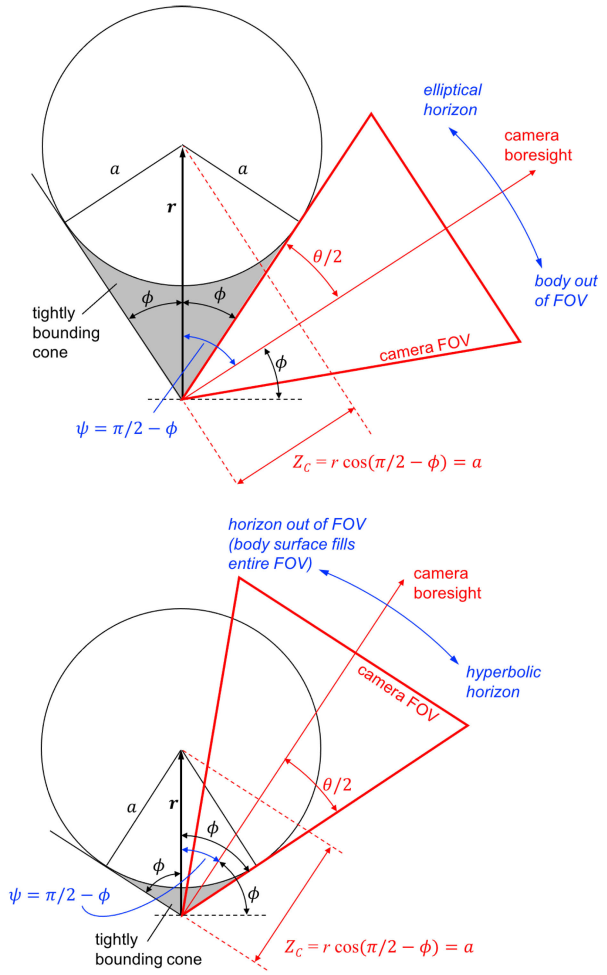

FIGURE 12. Limiting cases for an elliptical horizon (top) and hyperbolic horizon (bottom) with a camera of finite FOV. Only elliptical horizon arcs are possible for altitudes larger than the top scenario. Only hyperbolic horizon arcs are possible for altitudes smaller than the bottom scenario. The tightly bounding cone is shaded gray in both cases.

which, after recalling Eq. (50), leads to

$$\begin{aligned} A &= a^2 - Y_C^2 - Z_C^2 \\ B &= 2X_C Y_C \\ C &= a^2 - X_C^2 - Z_C^2 \end{aligned} \tag{65}$$

The apparent horizon is an ellipse when $B^2 - 4AC < 0$, so

$$B^2 - 4AC = (2X_C Y_C)^2 - 4(a^2 - Y_C^2 - Z_C^2)(a^2 - X_C^2 - Z_C^2) \tag{66}$$

$$B^2 - 4AC = 4(a^2 - Z_C^2)(\boldsymbol{r}_C^T \boldsymbol{r}_C - a^2) < 0 \tag{67}$$

Since the spacecraft must be outside the celestial body's sphere, we know that $\boldsymbol{r}_C^T \boldsymbol{r}_C > a^2$. Therefore, the projected horizon arc for a spherical body will be an ellipse when

$$a^2 - Z_C^2 < 0 \quad \rightarrow \quad a^2 < Z_C^2 \tag{68}$$

This analytic result agrees with simple geometric intuition, which may be seen in Fig. 12.

To enhance our intuitive understanding, we will consider this geometry a little further. Continuing with the temporary assumption of a spherical celestial body, the half angle ϕ of the tightly bounding cone is given by

$$\sin\phi = a/r \tag{69}$$

where $r = \|\boldsymbol{r}_C\|$.

We seek to determine if the projected horizon arc will be an ellipse or a hyperbola. The transition between these two cases is a parabolic horizon, which occurs when the angle between $\boldsymbol{r}_C$ and the image plane is the same as the cone half angle ϕ.

For a camera with a finite FOV, there is a finite range of altitudes where the parabolic horizon will be within the FOV. If the distance to the celestial body is too large, a parabolic horizon requires the body to be outside the camera FOV. Conversely, if the distance to the celestial body is too small, a parabolic horizon requires the body's surface completely fill the camera FOV. Figure 12 shows the limiting cases, which are now considered in more detail.

Suppose the camera has a FOV of θ. The top scenario from Fig. 12 produces the maximum range where a non-elliptical horizon arc is possible. The angle between $\boldsymbol{r}$ and the camera boresight, ψ, may be computed in two ways,

$$\psi_e = \frac{\pi}{2} - \phi_e = \phi_e + \frac{\theta}{2} \tag{70}$$

which may be solved for the limiting half-cone angle for the specified FOV

$$\phi_e = \frac{\pi}{4} - \frac{\theta}{4} \tag{71}$$

After substitution into Eq. (69), the limit on the half-cone angle yields a limit on the distance

$$r_e = a/\sin\phi_e = a\csc\left(\frac{\pi}{4} - \frac{\theta}{4}\right) \tag{72}$$

This result was presented without derivation in [47]. While correct and sufficient for OPNAV-only applications, this constraint alone is incomplete within the larger context of horizon-based navigation that also includes attitude determination in low-altitude orbits.

The minimum range where non-hyperbolic horizon arcs are possible may be computed by a similar procedure. Using the bottom scenario from Fig. 12, observe that

$$\psi_h = \frac{\pi}{2} - \phi_h = \phi_h - \frac{\theta}{2} \quad \rightarrow \quad \phi_h = \frac{\pi}{4} + \frac{\theta}{4} \tag{73}$$

$$r_h = a/\sin\phi_h = a\csc\left(\frac{\pi}{4} + \frac{\theta}{4}\right) \tag{74}$$

The bounds of Eqs. (72) and (74) are shown graphically in Fig. 13 using Earth as an example. The Earth limb is always an ellipse from geostationary orbit (GEO) and is usually a hyperbola from LEO. Similar results could be produced for the Moon, Mars, or any other celestial body.

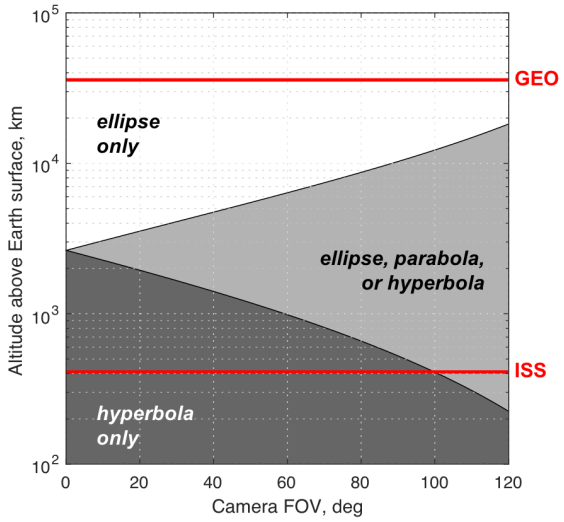

FIGURE 13. The types of horizon conics that are possible at different altitudes above a spherical Earth with radius 6,378 km. Red lines show the altitudes of the International Space Station orbit in LEO (about 410 km) and of a geostationary orbit (about 35,800 km).

4) REMARKS ON CIRCULAR HORIZON ARCS

It should be plainly evident at this point that horizon arcs are not generally circular—even for a spherical celestial body. Indeed, we have already shown that spherical bodies may produce hyperbolic horizon arcs when imaged from a low altitude. Despite this, there seems to be a widespread misconception that assuming circular horizon arcs for OPNAV is a good approximation and that such an approach has the benefit of "simplicity." Both claims are almost always false. The results of this tutorial clearly show there is no justification for the modern OPNAV practitioner to ever presuppose a circular horizon arc.

We begin our remarks by acknowledging that spherical bodies positioned along the camera boresight will produce a circular horizon arc. This may be intuitively deduced by taking a slice of a right circular cone with an image plane perpendicular the cone's axis of revolution (which, in the special case of a sphere, is also the direction from the camera to the body center). We may also arrive at this result algebraically by recalling that the projected conic is a circle when the coefficients of the implicit conic equation obey the constraint $(A - C)^2 + B^2 = 0$. For a spherical body, substitution from Eq. (65) shows that

$$(A - C)^2 + B^2 = \left(X_C^2 + Y_C^2\right)^2 = 0 \tag{75}$$

This implies that $X_C = 0$ and $Y_C = 0$, which means the vector from the camera to the celestial body center is $\boldsymbol{r}_C^T = [0, 0, Z_C]$ and is entirety along the camera boresight direction. Whenever the camera boresight is not pointed exactly at the body's center (for a sphere), the projected horizon arc will be something other than a circle (e.g., an ellipse).

For a spherical body having $Z_C > a$, the conic locus of the horizon is an ellipse given by Eq. (64). Some tedious algebra will show this horizon arc to have an ellipticity ($\epsilon = a_h/b_h$; ratio of semimajor a_h to semiminor axis b_h in image) of

$$\epsilon = a_h/b_h = \sqrt{\frac{r^2 - a^2}{Z_C^2 - a^2}} = \sqrt{\frac{\cos^2\phi}{\cos^2\psi - \sin^2\phi}} \tag{76}$$

which produces an eccentricity e of

$$e^2 = 1 - 1/\epsilon^2 = 1 - \frac{Z_C^2 - a^2}{r^2 - a^2} = \frac{\sin^2\psi}{\cos^2\phi} \tag{77}$$

where ϕ is the half-angle of the tightly bounding cone and ψ is the angle between $\boldsymbol{r}_C$ and the camera boresight direction. These angle definitions are the same as illustrated in Fig. 13. The ellipticity is often more extreme for non-spherical bodies. Moreover, we find the ellipticity of the projected horizon (even if small) to be critical for navigation performance and solution stability. The navigation analyst ignores this effect at their own peril.

F. HORIZON LOCALIZATION IN A DIGITAL IMAGE

For an image of a celestial body to be useful for navigation, we must be capable of automatically extracting the pixel coordinates of points of interest. In this work, we are interested in points belonging to the body's apparent horizon. Finding these points is an image processing task.

Less than half of the apparent horizon is usable for navigation in a typical image. Except in rare special cases, images of celestial bodies will be at non-zero phase angles and the terminator (day/night line) will be visible. As a consequence, part of the horizon will be illuminated (we call this the *lit limb*) and part will be in shadow. Only the lit limb is useful for navigation. A review of the example images from Fig. 8 will show the shadowed limb to be nearly undetectable (when it is detectable, the signal-to-noise ratio is poor). Moreover, we generally avoid using points along the lit limb near the cusps, where the shadow of the terminator begins to encroach on the lit horizon—the result is less than 180 deg of usable horizon arc in almost all practical cases. Past studies show the best localization performance is about ±70 deg from the limb's subsolar point (for a total of 140 deg of horizon arc) [48].

Before proceeding further, the reader is explicitly warned against using the entire lit limb (360 deg of arc) at very low phase angle, even though the celestial body's disk may qualitatively appear to be fully illuminated. In these cases, it is not uncommon for the terminator to lightly shadow (or slightly trim a few pixels off) one side of the apparent disk—even if it is indiscernible from casual inspection by the human analyst. The shadowed (or trimmed) horizon pixels will shift the apparent limb location towards the center of the body and corrupt the resulting navigation solution in a meaningful way. Outside of a simulation environment, superior performance is almost always achieved by restricting ourselves to only using horizon points belonging to the lit limb.

For an airless body (e.g., Moon, Pluto), there is a discontinuity in brightness across the lit limb. This discontinuity is spread out in a digital image due to the the optical

assembly's point spread function and spatial quantization by the FPA detector. Regardless, the underlying brightness discontinuity makes limb localization for an airless body equivalent to an edge localization problem. There are a variety of widely-available edge localization algorithms with pixel-level performance (i.e., these algorithms identify edge pixels), including: Sobel, Marr-Hilderth [49], Canny [50], and many others [51]. These pixel-level methods usually provide reasonable qualitative performance—making it possible to produce navigation solutions, compute statistics, and write papers. In practice, however, these pixel-level edge locations are inadequate for navigation. Horizon-based navigation using fits to image limb points *absolutely requires* subpixel horizon localization with contemporary (circa 2020) imaging sensors. Good results are achievable with the partial area effect (PAE) technique [52] and best-in-class performance is achievable with moment-based methods [5], [53]. The reader is directed to the image processing literature for additional details on the edge localization problem.

Regardless of the algorithm chosen, the end result is usually a set of n pixel coordinates along the celestial body's lit limb. We can describe this set of points in homogeneous coordinates as $\{\bar{\boldsymbol{u}}_i\}_{i=1}^{n}$.

Finally, the reader is cautioned against making undue use of the apparent terminator location. It is true that the terminator provides a direct measurement of phase angle, sometimes making it useful as a means of determining the direction to the Sun [54], [55]. However, because the terminator arc occurs where the incoming sunlight is tangent to the body's ellipsoidal surface, the sensitivity of terminator location to sunlight direction is quite poor. Further, unlike the lit limb, the terminator does not correspond to a physical discontinuity that manifests as an underlying discontinuity in image intensity. Instead, the terminator arc is a locus of points on the interior of the celestial body's apparent disk. Accurate localization of the terminator in an image is difficult because the transition from sunlight to shadow is gradual and often occurs over a very large number of pixels. Moreover, real bodies have texture that further complicates the fitting of simple models to the apparent terminator intensity gradient—specifically (1) globally varying surface albedo and (2) terrain relief that casts long shadows near the terminator. Until new techniques resolve these challenges, the combination of poor sensitivity to incoming sunlight direction and poor localization in real imagery makes the terminator a poor navigation observable for most applications.

G. IMAGING CELESTIAL BODIES AT DIFFERENT WAVELENGTHS

Most space imaging systems used for navigation operate in either the visible or infrared spectrum. It is often preferable to work in the visible spectrum since these cameras have higher resolution FPAs, produce less noisy images, are cheaper, and place fewer engineering requirements on the vehicle (e.g., thermal control). Observations in the visible spectrum work exceptionally well on airless bodies where there is a crisp and predictable horizon line.

For a celestial body with an atmosphere (e.g., Earth, Venus), there is not a brightness discontinuity across the lit limb when imaged in the visible spectrum. The scattering of sunlight in the atmosphere [56], [57] causes a gradual change in image brightness spanning the width of the atmosphere. Thus, in the visible spectrum, we find that the atmosphere obscures the true horizon and creates an apparent horizon at an unpredictable altitude above the surface. This is a long-standing challenge for horizon-based OPNAV and the specific issues are addressed at length elsewhere [5], [58]. The usual solution for robotic space exploration has been to ignore bodies with atmospheres and to only use images of airless bodies for horizon-based OPNAV, which has been satisfactory for past missions. While there has been some very preliminary studies of overcoming this problem for autonomous OPNAV [59], this early work is incomplete and suggests no specific path forward. This is an area ripe for future research.

Past spaceflight missions have not made extensive use of infrared images for horizon-based navigation because of (1) their lack of availability and (2) poor performance stemming from their comparatively low FPA resolution. For most planetary exploration applications—especially to airless bodies—visible spectrum images are preferred. Conversely, when we must use images of Earth, sensing in the infrared becomes a more attractive option. The Earth's atmosphere is known to possess a crisp horizon in the longwave infrared (LWIR) at wavelengths of 14.0–16.3 μm. This is due to infrared radiation emitted from the atmosphere's CO_2 band, which produces an apparent horizon at an altitude of about 30–50 km [14], [15]. For these reasons, most Earth horizon sensors used for attitude determination operate in the LWIR.

H. REMARKS ON CONIC FITTING IN AN IMAGE

Points lying on an ellipsoidal body's horizon form a conic in an image (see Sec. II-D). Thus, if we are able to find horizon points in an image (see Sec. II-F), it is sometimes useful to fit a conic to these measured horizon points. An illustrative example is shown in Fig. 14.

Horizon arcs are usually elliptical or hyperbolic (see Sec. II-E3), making circle fitting techniques (e.g., Taubin's method [61]) inappropriate for the present application. In the case where the spacecraft is always far away from the observed celestial body, the horizon will always be an ellipse (e.g., see Fig. 13) and ellipse-specific methods may be used in these situations. Otherwise, it is better to use generic conic fitting algorithms.

There are a variety of conic and ellipse fitting algorithms that have been developed over the last few decades. While a full accounting of these various methods is beyond the scope of the present work, there are both non-iterative (e.g., [60], [62]–[64]) and iterative methods (e.g., [65]–[67]). In the case where a large number of horizon points are observed over substantial horizon arcs, we find very little practical difference

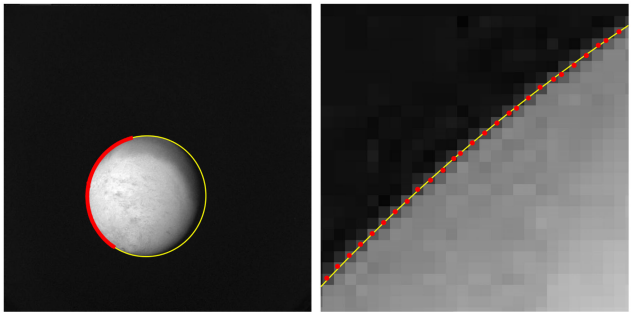

FIGURE 14. Example of conic fitting to horizon points on a real image of Triton (moon of Neptune). Pixel coordinates of points along the lit limb are found with subpixel accuracy using the algorithm from [53]. The "semihyper" least squares method from [60] is used to fit an ellipse to these image points. Left frame shows raw image from Voyager 2 spacecraft (image C1137709_CALIB, [38], [39]) with overlay of points on lit limb (red) and best-fit conic (yellow). Right frame shows zoom-in of lit limb so that individual subpixel horizon points may be seen (red dots).

between the best contemporary non-iterative methods (e.g., the "semihyper" least squares fit from [64] and [60]) and the latest iterative methods. The advantages of the iterative methods become more apparent when there are very few horizon points or when the arc size becomes small. These trends are supported by extensive numerical experiments throughout the remainder of this article and are consistent with the observations of [68].

Finally, note that all derivations in this tutorial work with conics in image plane coordinates instead of conics in pixel coordinates. The distinction between these two is discussed in Sec. II-D2. Since horizon localization algorithms usually provide a set of pixel coordinates $\{\bar{\boldsymbol{u}}_i\}_{i=1}^n$, there are two straightforward ways to obtain the desired image plane conic $\boldsymbol{C}$:

1) Transform $\{\bar{\boldsymbol{u}}_i\}_{i=1}^n$ to $\{\bar{\boldsymbol{x}}_i\}_{i=1}^n$ using Eq. (20). Apply a conic fitting algorithms to the points $\{\bar{\boldsymbol{x}}_i\}_{i=1}^n$ and directly obtain $\boldsymbol{C}$.
2) Apply a conic fitting algorithms to the points $\{\bar{\boldsymbol{u}}_i\}_{i=1}^n$ and directly obtain $\boldsymbol{C}_{uv}$. Transform $\boldsymbol{C}_{uv}$ to $\boldsymbol{C}$ using Eq. (54).

Regardless of the method chosen, we may easily find the conic in image plane coordinates. From this point forward we assume that any ellipse fit to measured horizon points may be written as either $\boldsymbol{C}$ (conic locus) or $\boldsymbol{C}^*$ (conic envelope).

III. OPTICAL NAVIGATION (OPNAV) WITH A WELL-KNOWN CELESTIAL BODY

The first of the three motivating scenarios from Sec. I is the case where the relative camera-to-celestial body attitude is known, but the relative position is unknown. This is exactly the situation when navigating relative to well-known celestial bodies and is the usual assumption for classical OPNAV. The objective here is to solve for the unknown location of the camera (and, hence, the location of spacecraft).

For many bodies of interest (e.g., Moon, Mars, Earth) we have excellent knowledge of both shape and orientation. For these well-known celestial bodies, we may obtain their attitude relative to an inertial frame (e.g., the International Celestial Reference Frame, ICRF [69]–[71]) from publicly available ephemeris files (e.g., SPICE kernels [72], [73]). Likewise, we usually know the spacecraft's inertial attitude through a star tracker [74], [75]. Therefore, if we know the celestial body's inertial attitude $\boldsymbol{T}_P^I$, the spacecraft inertial attitude $\boldsymbol{T}_S^I$, and the orientation of the camera on the spacecraft $\boldsymbol{T}_C^S$, we may find the relative attitude as

$$\boldsymbol{T}_C^P = \boldsymbol{T}_C^S \boldsymbol{T}_S^I \left(\boldsymbol{T}_P^I\right)^T = \boldsymbol{T}_C^S \boldsymbol{T}_S^I \boldsymbol{T}_I^P \tag{78}$$

This approach often produces relative attitude knowledge with errors less than a degree.

This form of horizon-based OPNAV has been used to navigate spacecraft for over 40 years [2]. Therefore, after a few brief historical remarks, we consider the geometry of spacecraft localization from an observed horizon arc in a digital image—ultimately arriving at the Christian-Robinson algorithm. We conclude with a detailed performance comparison, showing that the Christian-Robinson algorithm is both the simplest to implement and has the best performance.

A. HISTORICAL REMARKS ON OPICAL NAVIGATION

The concept of horizon-based navigation predates spaceflight by centuries. Early human explorers knew that the angles between the Earth's horizon and reference stars can be used to determine the observer's latitude and longitude (with the latter requiring a good way of measuring time). This was especially important for maritime exploration where there were few other sources of navigation information. Equipped with our geometric understanding from Sec. II-E, we recognize that these early explorers were finding the observer location that would place the apparent hyperbolic horizon arc in the right orientation relative to the inertially fixed starfield. This fact, however, was unknown to the explorers of antiquity, who developed their own empirical models for relating apparent star elevation measurements to observer location.

During the earliest days of space exploration, our centuries of history with maritime navigation using star-horizon measurements made horizon-based OPNAV an obvious thing to attempt. Numerous crew-based OPNAV experiments were performed during the Gemini, Apollo, and Skylab programs—ultimately demonstrating the efficacy of such an approach [76]–[80].

Concurrently with the advancement of the human spaceflight program in the 1960s, great advancements were also made on robotic space exploration. It was immediately evident that horizon-based measurements were equally valuable when navigating robotic spacecraft near other celestial bodies (e.g., Mars) as they were for crewed spacecraft near Earth. The difference is that robotic vehicles must take horizon measurements automatically with a camera (instead of manually by an astronaut). Interplanetary OPNAV was first demonstrated on images of Mars from the Mariner 6 and 7 missions in 1969 [81] and subsequently on Mariner 9 in 1971 [82]—paving the way for its first operational use during

the Voyager 1 flyby of Jupiter in 1979 [83]. Image-based OPNAV has been used to help navigate most subsequent outer planet exploration mission since then (e.g., Cassini [84], New Horizons [85]).

When an exploration spacecraft approaches a celestial body from far away, it is common for the celestial body to first appear as an unresolved object that subtends only a few image pixels (or less). Because cameras cannot directly measure depth, OPNAV algorithms at these large distances can only produce bearing (i.e., direction) measurements to the observed body. The task of finding this direction is often referred to as *centerfinding* since the image processing task is simply to find the centroid of the bright spot formed by the celestial body [3]. We also know from before (see Sec. II-A2) that specifying an object's center coordinates in an image describes a point in $\mathbb{P}^2$ and is equivalent to specifying the direction to that object. In this unresolved regime, precision OPNAV applications often require subpixel corrections for the bearing bias that occurs between the center of illumination and the celestial body's geometric center at non-zero phase angles. These subpixel corrections to the celestial body center (or bearing) may be computed analytically [86]–[88].

As the spacecraft gets closer, the celestial body will appear larger—eventually subtending a great number of image pixels. Once it is sufficiently large, the shape becomes apparent and it is possible to discern the apparent horizon. While the term *centerfinding* continues to be used in this context, doing so is often the source of confusion for two reasons. First, the center coordinates of the ellipse describing the celestial body's apparent horizon does not produce a vector pointing towards the center of the celestial body (i.e., the projection of $\boldsymbol{r}_C$ onto the image is not coincident with the center of the horizon ellipse). Second, the word ''centerfinding'' suggests that we need to find the center of the celestial body's apparent disk, when this is most certainly not the geometric quantity of interest. Thus, while the term centerfinding is appropriate for OPNAV with unresolved bodies, we suggest it be abandoned if favor of different language when discussing OPNAV with resolved bodies.

Once the body is fully resolved, most operational horizon-based OPNAV has been accomplished using the limb scan technique first developed for Voyager [3]. The limb scan technique is illustrated in Fig. 15 and notionally works as follows: Beginning with an initial guess of the body center and size, compute the predicted 1D brightness profile along scan lines emanating from the apparent body center. Compare the predicted brightness profile along each scan direction to the observed brightness profile along the same scan direction. Then, iteratively update the relative position between the spacecraft and the celestial body to obtain the best agreement between the predicted and observed brightness profiles. While this method has enjoyed great success over the years, it is (1) iterative and requires an initial guess, (2) requires a photometric model for the observed body to generate the predicted brightness profiles, (3) not guaranteed to always converge, and (4) known to produce estimates with a small bias. All four of these undesirable attributes may be solved by making use of the geometric results introduced in Sec. II, with best results coming from the Christian-Robinson algorithm. Comparative studies on historical flight data demonstrates the efficacy of these new methods in practice [48].

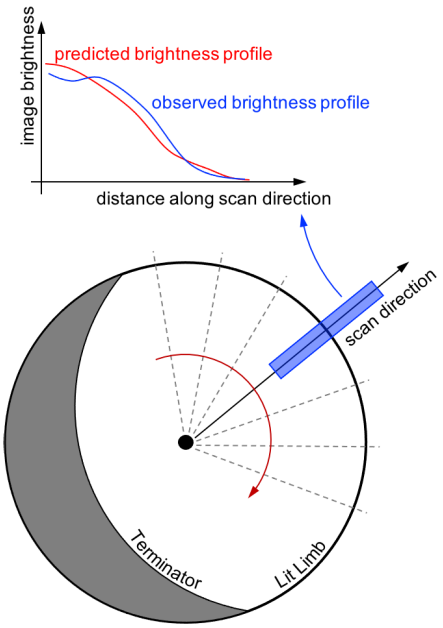

FIGURE 15. Horizon-based OPNAV with ellipsoidal celestial bodies has historically been accomplished by minimizing the residuals between observed and predicted brightness profiles along a collection of one-dimensional scan lines in the image. While still widely used due to decades of flight heritage, this technique is outdated and no longer represents the state-of-the-art.

B. OPNAV BY ELLIPTICAL HORIZON FITTING

The most obvious approach for horizon-based OPNAV is to fit an ellipse to the horizon points and then relate the cone formed by the back projection of this ellipse to the celestial body's tightly-bounding cone, as is done Eq. (59). This approach was taken in [46] and results in a procedure that directly estimates the unknown relative position $\boldsymbol{r}_C$. Unfortunately, while this certainly works, OPNAV by ellipse fitting has comparatively poor performance in the presence of measurement noise and is now an outdated method. The method is briefly reviewed here for completeness (and for important parallels to the algorithms in Sec. IV and Sec. V), but the reader is strongly advised against taking this approach for practical OPNAV. Instead, the Christian-Robinson algorithm from Sec. III-C is almost always the better choice.

Therefore, following the insights from [46], we begin by relating the image conic and the tightly bounding cone. This can be done by equating either the quadric cones in $\mathbb{P}^3$ or their projections in an image in $\mathbb{P}^2$. Either way, one obtains the familiar relation from Eq. (59)

$$\boldsymbol{C} \propto \boldsymbol{M}_C = \boldsymbol{A}_C \boldsymbol{r}_C \boldsymbol{r}_C^T \boldsymbol{A}_C - (\boldsymbol{r}_C^T \boldsymbol{A}_C \boldsymbol{r}_C - 1)\boldsymbol{A}_C \qquad (79)$$

Right-multiplying by $\boldsymbol{r}_C$ reveals that

$$\boldsymbol{C}\boldsymbol{r}_C \propto \boldsymbol{A}_C\boldsymbol{r}_C\boldsymbol{r}_C^T\boldsymbol{A}_C\boldsymbol{r}_C - (\boldsymbol{r}_C^T\boldsymbol{A}_C\boldsymbol{r}_C - 1)\boldsymbol{A}_C\boldsymbol{r}_C \tag{80}$$

and the right-hand side simplifies to yield

$$\boldsymbol{C}\boldsymbol{r}_C \propto \boldsymbol{A}_C\boldsymbol{r}_C \tag{81}$$

Therefore, since we know

$$\det(\boldsymbol{A}_C) = \det(\boldsymbol{A}_P) = (abc)^{-2} \not\approx 0 \tag{82}$$

we can always compute $\boldsymbol{A}_C^{-1}$ and it becomes apparent that

$$\boldsymbol{r}_C \propto \boldsymbol{A}_C^{-1}\boldsymbol{C}\boldsymbol{r}_C \tag{83}$$

Or, after removing the arbitrary scaling,

$$\lambda\boldsymbol{e}_C = \boldsymbol{A}_C^{-1}\boldsymbol{C}\boldsymbol{e}_C \tag{84}$$

which is simply a 3×3 eigenvalue/eigenvector problem. It follows that the direction to the body center $\boldsymbol{e}_C = \boldsymbol{r}_C/\|\boldsymbol{r}_C\|$ must be one of the unit eigenvectors of the square matrix $\boldsymbol{A}_C^{-1}\boldsymbol{C}$. The matrix $\boldsymbol{A}_C^{-1}\boldsymbol{C}$ is guaranteed to be indefinite, having two eigenvalues of one sign and one eigenvalue of the other sign. As discussed in [46], the unit eigenvector we seek is the one corresponding to the eigenvalue of unique sign.

To find the range $r = \|\boldsymbol{r}_C\|$, we substitute $\boldsymbol{r}_C = r\boldsymbol{e}_C$ into Eq. (79) along with the scaling $\boldsymbol{C} = \lambda\boldsymbol{M}_C$ to find

$$\boldsymbol{C} = \lambda\left[r^2\boldsymbol{A}_C\boldsymbol{e}_C\boldsymbol{e}_C^T\boldsymbol{A}_C - (r^2\boldsymbol{e}_C^T\boldsymbol{A}_C\boldsymbol{e}_C - 1)\boldsymbol{A}_C\right] \tag{85}$$

where λ is the eigenvalue of $\boldsymbol{A}_C^{-1}\boldsymbol{C}$ having a unique sign. Taking the trace and solving for r^2 yields the result for range from [46]:

$$r^2 = \frac{Tr[\boldsymbol{C}] - \lambda Tr[\boldsymbol{A}_C]}{\lambda\boldsymbol{e}_C^T(\boldsymbol{A}_C\boldsymbol{A}_C - Tr[\boldsymbol{A}_C]\boldsymbol{A}_C)\boldsymbol{e}_C} \tag{86}$$

While this result is exact in the noise-free case, performance degrades quickly in the presence of noise. It is better to use the Christian-Robinson algorithm.

C. THE CHRISTIAN-ROBINSON ALGORITHM

The Christian-Robinson algorithm provides a means of directly solving for $\boldsymbol{r}_C$ from the observed horizon points without requiring an intermediate conic fit. The idea is to factor the celestial body's symmetric shape matrix $\boldsymbol{A}_C$ as

$$\boldsymbol{A}_C = \boldsymbol{B}^T\boldsymbol{B} \tag{87}$$

which may be used to transform the generic problem (which is complicated) into a trivial problem. This approach was first proposed by Christian and Robinson in [89], where we used a Cholesky factorization to obtain $\boldsymbol{B}$. The idea was subsequently improved upon in [5] by use of a more natural factorization. Informed by a few years of practical use, the idea reaches full maturity in the present discussion.

To begin, substitute Eq. (59) into Eq. (49) to find

$$\bar{\boldsymbol{x}}_i^T\left[\boldsymbol{A}_C\boldsymbol{r}_C\boldsymbol{r}_C^T\boldsymbol{A}_C - (\boldsymbol{r}_C^T\boldsymbol{A}_C\boldsymbol{r}_C - 1)\boldsymbol{A}_C\right]\bar{\boldsymbol{x}}_i = 0 \tag{88}$$

where we now explicitly keep track of different horizon points $\bar{\boldsymbol{x}}_i$ by introduction of the subscript i. From here, substitute the factorization of Eq. (87) for $\boldsymbol{A}_C$,

$$\bar{\boldsymbol{x}}_i^T\left[\boldsymbol{B}^T\boldsymbol{B}\boldsymbol{r}_C\boldsymbol{r}_C^T\boldsymbol{B}^T\boldsymbol{B} - (\boldsymbol{r}_C^T\boldsymbol{B}^T\boldsymbol{B}\boldsymbol{r}_C - 1)\boldsymbol{B}^T\boldsymbol{B}\right]\bar{\boldsymbol{x}}_i = 0 \tag{89}$$

which is simplified by pulling out a $\boldsymbol{B}^T$ to the left and a $\boldsymbol{B}$ to the right

$$(\boldsymbol{B}\bar{\boldsymbol{x}}_i)^T\left[(\boldsymbol{B}\boldsymbol{r}_C)(\boldsymbol{B}\boldsymbol{r}_C)^T - \left((\boldsymbol{B}\boldsymbol{r}_C)^T(\boldsymbol{B}\boldsymbol{r}_C) - 1\right)\right](\boldsymbol{B}\bar{\boldsymbol{x}}_i) = 0 \tag{90}$$

Now, use the factor $\boldsymbol{B}$ to define $\bar{\boldsymbol{x}}_i'$ and $\boldsymbol{r}'$ in a transformed (primed) space,

$$\bar{\boldsymbol{x}}_i' = \boldsymbol{B}\bar{\boldsymbol{x}}_i \tag{91}$$

$$\boldsymbol{r}' = \boldsymbol{B}\boldsymbol{r}_C \tag{92}$$

such that Eq. (90) simplifies to

$$\bar{\boldsymbol{x}}_i'^T\left[\boldsymbol{r}'\boldsymbol{r}'^T - \left(\boldsymbol{r}'^T\boldsymbol{r}' - 1\right)\right]\bar{\boldsymbol{x}}_i' = 0 \tag{93}$$

Therefore, in the transformed space, we see every problem becomes navigation relative to a unit sphere. To make this explicit, we can introduce the 3×3 identity matrix $\boldsymbol{I}_3$ in a few locations to find

$$\bar{\boldsymbol{x}}_i'^T\left[\boldsymbol{I}_3\boldsymbol{r}'\boldsymbol{r}'^T\boldsymbol{I}_3 - \left(\boldsymbol{r}'^T\boldsymbol{I}_3\boldsymbol{r}' - 1\right)\boldsymbol{I}_3\right]\bar{\boldsymbol{x}}_i' = 0 \tag{94}$$

which is exactly Eq. (88) with $\boldsymbol{A}_C = \boldsymbol{I}_3$. From Eqs. (24) and (28), it is clear that $\boldsymbol{A}_C = \boldsymbol{I}_3$ describes the unit sphere.

We briefly observe that horizon localization algorithms natively provide measurements in pixel coordinates and not image plane coordinates (i.e., we are given $\bar{\boldsymbol{u}}_i$ instead of $\bar{\boldsymbol{x}}_i$). This distinction not especially important for a calibrated camera, since it easy to go between the two using Eq. (19) and Eq. (20). It is, however, computationally convenient to combine the transformation from $\bar{\boldsymbol{u}}_i$ to $\bar{\boldsymbol{x}}_i$ with the transformation from $\bar{\boldsymbol{x}}_i$ to $\bar{\boldsymbol{x}}_i'$. Thus, substituting Eq. (20) into Eq. (91),

$$\bar{\boldsymbol{x}}_i' = \boldsymbol{B}\bar{\boldsymbol{x}}_i = \boldsymbol{B}\boldsymbol{K}^{-1}\bar{\boldsymbol{u}}_i \equiv \boldsymbol{R}\bar{\boldsymbol{u}}_i \tag{95}$$

where the combined transformation is given by the 3×3 matrix $\boldsymbol{R}$

$$\boldsymbol{R} = \boldsymbol{B}\boldsymbol{K}^{-1} \tag{96}$$

Knowing that a factorization of $\boldsymbol{A}_C = \boldsymbol{B}^T\boldsymbol{B}$ transforms every horizon-based OPNAV problem to navigation relative to the unit sphere, the Christian-Robinson algorithm has two components. The first component is to find the most convenient form of $\boldsymbol{B}$. The second component is to solve the unit sphere navigation problem in closed form. These components are now presented, followed by a discussion of how they may be integrated into a solution for the fully general horizon-based OPNAV problem.

1) FACTORIZATION OF THE BODY SHAPE MATRIX

Consider first the celestial body's shape matrix in the body's own principal axis frame, $\boldsymbol{A}_P$ [see Eq. (24)]. Factorization in this case is trivial

$$\boldsymbol{A}_P = \boldsymbol{D}^T\boldsymbol{D} \tag{97}$$

where $\boldsymbol{D}$ is the 3×3 diagonal matrix

$$\boldsymbol{D} = \begin{bmatrix} 1/a & 0 & 0 \\ 0 & 1/b & 0 \\ 0 & 0 & 1/c \end{bmatrix} \tag{98}$$

Substituting this result into Eq. (28) and grouping like terms

$$\boldsymbol{A}_C = \boldsymbol{T}_C^P\boldsymbol{A}_P\boldsymbol{T}_P^C = \boldsymbol{T}_C^P\boldsymbol{D}^T\boldsymbol{D}\boldsymbol{T}_P^C \tag{99}$$

$$= \left(\boldsymbol{D}\boldsymbol{T}_P^C\right)^T\boldsymbol{D}\boldsymbol{T}_P^C = \boldsymbol{B}^T\boldsymbol{B} \tag{100}$$

which produces the following simple expression for $\boldsymbol{B}$

$$\boldsymbol{B} = \boldsymbol{D}\boldsymbol{T}_P^C \tag{101}$$

It is trivial to compute $\boldsymbol{B}$ from Eq. (101) in practice since both $\boldsymbol{D}$ and $\boldsymbol{T}_P^C$ are known (the former since shape parameters a, b, c are known and the latter since relative attitude is assumed known in the classic OPNAV scenario).

Likewise, it is trivial to compute the combined transformation matrix $\boldsymbol{R}$ by substitution of Eq. (101) into Eq. (96)

$$\boldsymbol{R} = \boldsymbol{D}\boldsymbol{T}_P^C\boldsymbol{K}^{-1} \tag{102}$$

since $\boldsymbol{D}$ and $\boldsymbol{T}_P^C$ are known (as above) and $\boldsymbol{K}^{-1}$ is known from Eq. (21) for a calibrated camera.

2) OPNAV WITH A UNIT SPHERE

The transformation in Eqs. (91) and (92) converts every horizon-based OPNAV problem to navigation relative to a unit sphere. This is true even if the celestial body is a triaxial ellipsoidal. Simple geometry allows navigation with respect to a unit sphere to be solved exactly.

Consider the geometry from Fig. 16. Letting $r' = \|\boldsymbol{r}'\|$, we see that the tightly bounding cone in the transformed space has a half angle of ϕ'

$$\sin\phi' = 1/r' \tag{103}$$

This, of course, is the same as Eq. (69), but in the transformed space where an arbitrary ellipsoidal body has become a unit sphere. Now, also observe that the points $\bar{\boldsymbol{x}}_i'$ are computed from the measured horizon points [see Eq. (95)] and correspond to rays lying on the the tightly bounding cone. Since the transformed body is a sphere, the tightly bounding cone is a right circular cone (this is not generally true for the tightly bounding cone of an ellipsoidal body) and the angle between the cone centerline and the horizon rays is constant. That is,

$$\cos\phi' = \frac{\bar{\boldsymbol{x}}_i'^T\boldsymbol{r}'}{\|\bar{\boldsymbol{x}}_i'\|\,\|\boldsymbol{r}'\|}, \quad \forall i \tag{104}$$

It becomes convenient, therefore, to turn the transformed observations $\bar{\boldsymbol{x}}_i'$ into unit vector directions,

$$\boldsymbol{s}_i' = \bar{\boldsymbol{x}}_i'/\|\bar{\boldsymbol{x}}_i'\| \tag{105}$$

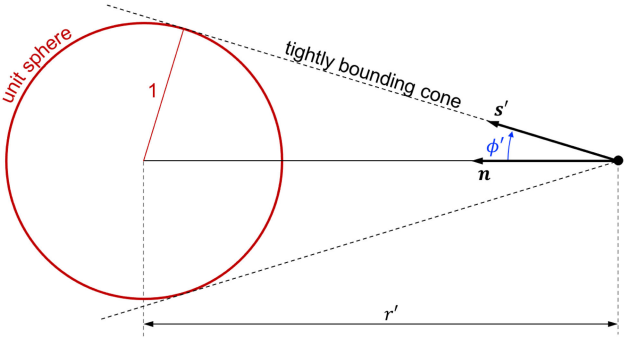

FIGURE 16. Illustration of geometry for a navigation near a unit sphere.

Having done this, we may rewrite Eq. (104) as

$$1 = \boldsymbol{s}_i'^T\boldsymbol{n} \tag{106}$$

where the new variable $\boldsymbol{n}$ is

$$\boldsymbol{n} = \frac{1}{r'\cos\phi'}\boldsymbol{r}' = \frac{1}{\cos\phi'}\boldsymbol{e}' \tag{107}$$

and where $\boldsymbol{e}' = \boldsymbol{r}'/\|\boldsymbol{r}'\| = \boldsymbol{n}/\|\boldsymbol{n}\|$. Thus the vector $\boldsymbol{n}$ is in the direction of $\boldsymbol{r}'$ and has a length that is functionally related to the distance between the camera and body center. We will return to finding the distance in a moment. First, however, we observe that for many transformed horizon measurements $\bar{\boldsymbol{x}}_i'$ we may stack Eq. (106) for each observation in matrix–vector notation to obtain the following linear system

$$\boldsymbol{H}\boldsymbol{n} = \boldsymbol{1}_{n\times 1} \tag{108}$$

where

$$\boldsymbol{H} = \begin{bmatrix} \boldsymbol{s}_1'^T \\ \boldsymbol{s}_2'^T \\ \vdots \\ \boldsymbol{s}_n'^T \end{bmatrix} \tag{109}$$

This linear system may be solved for $\boldsymbol{n}$ in either the least squares sense or the total least squares (TLS) sense [90], [91]. The TLS solution is attractive since the matrix $\boldsymbol{H}$ is noisy.

The next step is to convert $\boldsymbol{n}$ to $\boldsymbol{r}'$. To solve, observe that

$$\|\boldsymbol{n}\|^2 = \boldsymbol{n}^T\boldsymbol{n} = \frac{1}{\cos^2\phi'} = \frac{1}{1-\sin^2\phi'} \tag{110}$$

Using Eq. (103), substitute for $\sin^2\phi'$ in the right-hand term

$$\|\boldsymbol{n}\|^2 = \boldsymbol{n}^T\boldsymbol{n} = \frac{1}{1-1/r'^2} = \frac{r'^2}{r'^2-1} \tag{111}$$

Which may be solved for r'

$$r' = \sqrt{\frac{\boldsymbol{n}^T\boldsymbol{n}}{\boldsymbol{n}^T\boldsymbol{n}-1}} \tag{112}$$

Therefore, writing $\boldsymbol{r}'$ in terms of r' and $\boldsymbol{e}'$,

$$\boldsymbol{r}' = r'\boldsymbol{e}' = \frac{r'}{\|\boldsymbol{n}\|}\boldsymbol{n} \tag{113}$$

Now, recalling that $\|\boldsymbol{n}\| = \sqrt{\boldsymbol{n}^T\boldsymbol{n}}$ and substituting for r' from Eq. (112),

$$\boldsymbol{r}' = \left(\boldsymbol{n}^T\boldsymbol{n} - 1\right)^{-1/2}\boldsymbol{n} \tag{114}$$

3) RECOVERING OPNAV SOLUTION FOR A GENERIC BODY

The final step in the Christian-Robinson algorithm is to transform $\boldsymbol{r}'$ back to $\boldsymbol{r}_C$. This may be done by inverting the expression from Eq. (92),

$$\boldsymbol{r}_C = \boldsymbol{B}^{-1}\boldsymbol{r}' \tag{115}$$

It is possible to compute $\boldsymbol{B}^{-1}$ directly (i.e., without having to invert a generic matrix) since

$$\boldsymbol{B}^{-1} = \left(\boldsymbol{D}\boldsymbol{T}_P^C\right)^{-1} = \boldsymbol{T}_C^P\boldsymbol{D}^{-1} \tag{116}$$

and where $\boldsymbol{D}^{-1}$ is trivial to compute since $\boldsymbol{D}$ is diagonal [see Eq. (98)],

$$\boldsymbol{D}^{-1} = \begin{bmatrix} a & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & c \end{bmatrix} \tag{117}$$

Therefore, substituting $\boldsymbol{B}^{-1} = \boldsymbol{T}_C^P\boldsymbol{D}^{-1}$ into Eq. (115),

$$\boldsymbol{r}_C = \boldsymbol{T}_C^P\boldsymbol{D}^{-1}\boldsymbol{r}' \tag{118}$$

and, after substituting for $\boldsymbol{r}'$ in terms of $\boldsymbol{n}$ from Eq. (114),

$$\boldsymbol{r}_C = \left(\boldsymbol{n}^T\boldsymbol{n} - 1\right)^{-1/2}\boldsymbol{T}_C^P\boldsymbol{D}^{-1}\boldsymbol{n} \tag{119}$$

which provides a direct solution for $\boldsymbol{r}_C$ as a function of the least-squares solution for $\boldsymbol{n}$ from Eq. (108).

4) SUMMARY OF CHRISTIAN-ROBINSON ALGORITHM

Despite the rather lengthy derivation, the final implementation of Christian-Robinson algorithm is very simple—often consisting of only a few lines of code. The complete computational procedure is summarized in Algorithm 1. The reader will note that this algorithm is non-iterative and exact under the assumption of a pinhole camera model and an ellipsoidal celestial body. The reader will also note that no ellipse fit is performed in the Christian-Robinson algorithm, which results in significant computational savings as compared to outdated methods that attempt to explicitly fit an ellipse to points in the image. A version of this algorithm has been selected as part of the Orion OPNAV system that will be demonstrated during the Artemis I mission.

Algorithm 1 Pseudocode for the Christian-Robinson OPNAV algorithm

procedure $\boldsymbol{r}_C = \text{OPNAV}(\{\bar{\boldsymbol{u}}_i\}_1^n, \boldsymbol{K}^{-1}, \boldsymbol{T}_P^C, a, b, c)$
  compute $\boldsymbol{D} = \text{diag}[1/a, 1/b, 1/c]$ ▷ Eq. (98)
  compute $\boldsymbol{R} = \boldsymbol{D}\boldsymbol{T}_P^C\boldsymbol{K}^{-1}$ ▷ Eq. (102)
  for $i = 1$ to n **do**
    $\bar{\boldsymbol{x}}_i' = \boldsymbol{R}\bar{\boldsymbol{u}}_i$ ▷ Eq. (95)
    $\boldsymbol{s}_i' = \bar{\boldsymbol{x}}_i'/\|\bar{\boldsymbol{x}}_i'\|$ ▷ Eq. (105)
  construct $\boldsymbol{H}$ from $\{\boldsymbol{s}_i'\}_{i=1}^n$ ▷ Eq. (109)
  compute TLS solution for $\boldsymbol{n}$ ▷ Eq. (108)
  compute $\boldsymbol{T}_C^P = (\boldsymbol{T}_P^C)^T$
  compute $\boldsymbol{D}^{-1} = \text{diag}[a, b, c]$ ▷ Eq. (117)
  compute $\boldsymbol{r}_C$ ▷ Eq. (119)
return $\boldsymbol{r}_C$

D. OPNAV PERFORMANCE COMPARISON

The Christian-Robinson algorithm will always outperform methods that first fit a conic (usually an ellipse) to the horizon points and then attempt to compute the camera location from the best-fit conic. In general, the problem with fitting an ellipse first is that the ellipse fitting algorithms do not constrain the fit to be a valid projection of the celestial body. Specifically, the typical ellipse fitting algorithm admits five degrees-of-freedom in the fit, when the OPNAV problem only has three degrees-of-freedom. Thus, if we fit the ellipse first, we try to find the camera position that produces a horizon that is as close as possible to the observed ellipse [e.g., with Eq. (84)]. This is generally different than (and not as good as) solving for the camera position directly. This point is best illustrated through a numerical example.

Consider the case of viewing the Moon (radius of about 1,737 km) from a range of 25,000 km. Suppose we image the Moon with a camera having a 20 deg FOV and a $2,048 \times 2,048$ pixel FPA (a 4.2 megapixel image). Let the angle between the direction to the Moon and the camera boresight be $\psi = 8$ deg. Assuming a subpixel horizon localization algorithm (e.g., Ref. [53]), let the pixel errors on the horizon points be $\sigma = 0.07$ pixel. Performing a 10,000-run Monte Carlo results in the OPNAV performance statistics summarized in Table 2 and Table 3. In all cases, we observe from Table 3 that the uncertainty is orders of magnitude larger in the direction of the observed celestial body—which, for modest FOV cameras, is mostly along the boresight (z-axis) direction. This is consistent with analytic OPNAV covariance studies [92].

Looking at the data in Table 2, we see immediately that the Christian-Robinson algorithm is better than any result depending on a circle or ellipse fit. Some additional observations about the other algorithms are now made.

TABLE 2. OPNAV performance statistics for synthetic images of the Moon. Statistics are computed from a 10,000-run Monte Carlo simulation, with each algorithm operating on the same set of simulation inputs.

Method	Mean Error (km)	Standard Deviation (km)
Circle Methods		
Taubin Circle Fit [61]	232.7594	0.5256
Non-Iterative Ellipse Methods		
Modified Fitzgibbon [62], [63]	0.4481	3.8202
Semihyper Least-Squares [60]	0.0584	3.8205
Hyper Least-Squares [60]	0.0584	3.8205
Iterative Ellipse Methods		
Doubly-Optimal Fit [66]	0.0587	3.8204
Fund. Numerical Scheme [65]	0.0576	3.8204
Szpak, et al. [67]	0.0233	3.8204
Christian-Robinson Alg. [5]	0.0074	0.5311

TABLE 3. OPNAV covariance for synthetic images of the Moon. The root-sum-square (RSS) of these values are equivalent to the right-most column in Table 2. Errors are expressed in the camera frame.

Method	σ_x (km)	σ_y (km)	σ_z (km)
Circle Methods			
Taubin Circle Fit [61]	0.0282	0.0203	0.5245
Non-Iterative Ellipse Methods			
Modified Fitzgibbon [62], [63]	0.1277	0.0847	3.8171
Semihyper Least-Squares [60]	0.1277	0.0847	3.8174
Hyper Least-Squares [60]	0.1277	0.0847	3.8174
Iterative Ellipse Methods			
Doubly-Optimal Fit [66]	0.1277	0.0847	3.8173
Fund. Numerical Scheme [65]	0.1277	0.0847	3.8173
Szpak, et al. [67]	0.1277	0.0847	3.8173
Christian-Robinson Alg. [5]	0.0283	0.0203	0.5299

The problem set-up described above produces an ellipse in the image with a semimajor axis of 412.5 pixels and a semiminor axis of 408.5 pixels. Thus, the horizon arc is clearly not a circle. This significant model mismatch (circle fit of an elliptical horizon) accounts for the very large bias in the estimate of $\boldsymbol{r}_C$ when using a Taubin circle fit. The standard deviation with a circle fit is similar to that of the Christian-Robinson algorithm. Thus, the circle fit approach provides estimates of $\boldsymbol{r}_C$ with good precision but poor accuracy. Given the comparable algorithmic complexity of the Taubin circle fit (or similar) to the ellipse fits, there is no compelling reason to ever use a circle fit for horizon-based OPNAV.

Observe that there is little practical difference between the analytic "semihyper least-squares" solution and the more complicated iterative solutions. This is generally the case when we have a large number of horizon points (as we often do when the celestial body subtends a substantial portion of the image) with low noise (as we do with subpixel limb localization). Similar observations were made by Szpak, *et al.*, in [68]. Thus, if an ellipse fit must be used (not recommended), the semihyper ellipse fit from [60] is likely the best choice for onboard navigation applications.

Not quantitatively evaluated here are results from older algorithms that iteratively solve for $\boldsymbol{r}_C$ given an initial guess. As discussed above, there are two major types of these algorithms. The first is the legacy limb-scan technique that has been used operationally for over 40 years [3]. The limb-scan method has been shown to produce a similar standard deviation to the Christian-Robinson algorithm, but is known to have a bias in the estimate of $\boldsymbol{r}_C$ [48]. The second are whole-limb fit techniques where the horizon is reprojected onto the image and the position $\boldsymbol{r}_C$ is iteratively updated. The iterative whole-limb fit methods use gradient-based optimization methods to minimize residuals such as the distance between the horizon and limb points [47] or the difference in the gradient across the limb boundary [4]. Experience has shown these iterative methods to produce similar OPNAV results as the Christian-Robinson algorithm (when they converge, which is not guaranteed), but at the expense of considerably more algorithmic complexity. Therefore, the Christian-Robinson algorithm is usually the preferred method for computing $\boldsymbol{r}_C$ from observations of an ellipsoidal body's lit limb.

IV. ATTITUDE DETERMINATION

When the relative position is known, a celestial body's apparent horizon in an image may be used to compute the relative attitude. This is the second of the three motivating scenarios from Sec. I. While this problem has been well-studied, there seems to be some confusion on when different types of attitude determination problems arise in practice. The fundamental question is which object has the unknown attitude (the celestial body or the spacecraft) and in what frame is the relative position known (if it is known, it will be generally be known in the frame of the object with known attitude). These two scenarios are now explained.

Scenario 1: For a spacecraft in Earth orbit, there are a variety of means for determining the orbit in an Earth-fixed frame. This may be done with ground-based tracking, an onboard GPS receiver, or a number of other means. In such a case, we would know the position of the spacecraft relative to the Earth's principal axis frame. Since we know the Earth's attitude (both inertially and within it's own principal axis frame), the spacecraft must be the object with unknown attitude if there is to be an attitude determination problem to solve. This is exactly the problem solved by Earth horizon sensors [14]–[17]. In this case we would know $\boldsymbol{r}_P$ but have no knowledge of $\boldsymbol{T}_C^P$.

Scenario 2: When visiting a celestial body other than Earth, we are unlikely to natively know the position vector in the celestial body's principal axis frame. If the position vector from the spacecraft to the celestial body is known from sources other than the spacecraft imaging system, it is almost certainly known in some inertial frame (e.g., ICRF and obtained by differencing the known orbits of the celestial body and the spacecraft). Moreover, if a spacecraft has successfully navigated from Earth to another celestial body, it is reasonable to assume that the spacecraft has a means of independently determining it's inertial attitude (e.g., with a star tracker). Thus, the inertial position vector may be expressed in the camera frame because we know the attitude of the spacecraft. If an attitude determination problem exists, then it occurs because the celestial body's attitude is unknown. For these reasons, the scenario where horizon-based attitude determination makes the most sense for interplanetary exploration is when the celestial body has an unknown (or unpredictable) attitude. In this case we would know $\boldsymbol{r}_C$ but have no knowledge of $\boldsymbol{T}_C^P$.

These two scenarios are now discussed in more detail. Attitude determination algorithms are developed for both using a framework consistent with the OPNAV development in Sec. III.

A. FINDING SPACECRAFT ATTITUDE (POSITION KNOWN IN THE CELESTIAL BODY FRAME, P)

The first attitude determination scenario of interest is when the the relative position is known in the celestial body's

principal axis frame. If the position is known the the celestial body's frame, then it is the spacecraft whose attitude is unknown. This is most commonly the case for Earth orbiting spacecraft and is the type of problem solved by Earth horizon sensors.

The simplest geometry occurs when the observed body is a perfect sphere, though the full three degree-of-freedom attitude is not observable in this case. More interesting is the situation when the observed body is either an oblate spheroid or a triaxial ellipsoid. In both of these cases (spheroid and ellipsoid) the full three degree-of-freedom attitude may determined up to a twofold ambiguity.

1) THE SPECIAL CASE OF ATTITUDE DETERMINATION WITH A SPHERE

Suppose we know the relative position in the celestial body's principal axis frame, $\boldsymbol{r}_P$. When the observed celestial body is assumed to be a sphere, the full attitude is unobservable. Since $\boldsymbol{A}_C = \boldsymbol{A}_P = a^{-2}\boldsymbol{I}_3$ for a spherical body, we may presume the matrix $\boldsymbol{A}_C$ is known—and, with $\boldsymbol{A}_C$ known, we may use the Christian-Robinsion algorithm (see Sec. III-C) to compute $\boldsymbol{r}_C$. Thus, we know the relative position in both the celestial body's frame $\boldsymbol{r}_P$ and in the camera frame $\boldsymbol{r}_C$.

Knowing a single vector in two different frames permits determination of two components of the attitude, with rotation about the vector's direction being the one remaining unobservable component of attitude. Rather than obtaining an instantaneous attitude estimate, pairs of the same vector expressed in two different frames (in this case, $\boldsymbol{r}_P$ and $\boldsymbol{r}_C$ or their corresponding unit vectors $\boldsymbol{e}_P$ and $\boldsymbol{e}_C$) may be processed within an attitude filter (e.g., a multiplicative extended Kalman filter [93], [94]).

2) GENERIC ATTITUDE DETERMINATION WITH THE CONIC ENVELOPE: MODENINI's SOLUTION

Again suppose we know the relative position in the celestial body's principal axis frame, $\boldsymbol{r}_P$. Now, however, assume that the observed celestial body is either a spheroid or a triaxial ellipsoid. In this situation, recent work by Modenini [95] suggests an elegant way of thinking about the spacecraft attitude determination problem from projected horizons. Modenini's approach, however, leaves room for some algorithmic improvements. Therefore, after a brief review of Modenini's solution we suggest a few minor improvements for solving this problem.

Modenini begins with Eq. (61) and introduces a scalar α to account for the proportionality relationship

$$\alpha \boldsymbol{C}^* = \boldsymbol{A}_C^{-1} - \boldsymbol{r}_C \boldsymbol{r}_C^T \tag{120}$$

Since we are considering the case where position is known in the celestial body's frame, we know $\boldsymbol{r}_P$ and not $\boldsymbol{r}_C$. Therefore, rewrite the right-hand side of Eq. (120) in the celestial body's principal axis frame,

$$\alpha \boldsymbol{C}^* = \boldsymbol{T}_C^P \left(\boldsymbol{A}_P^{-1} - \boldsymbol{r}_P \boldsymbol{r}_P^T \right) \boldsymbol{T}_P^C \tag{121}$$

To compact notation, let $\boldsymbol{B}_M^* = \boldsymbol{A}_P^{-1} - \boldsymbol{r}_P \boldsymbol{r}_P^T$ to arrive at

$$\alpha \boldsymbol{C}^* = \boldsymbol{T}_C^P \boldsymbol{B}_M^* \boldsymbol{T}_P^C \tag{122}$$

where we note that $\boldsymbol{C}^*$ is measured and $\boldsymbol{B}_M^*$ is assumed known. Hence, it is possible to solve for the scalar α by taking the trace of both sides

$$\alpha = \frac{Tr\left[\boldsymbol{B}_M^*\right]}{Tr\left[\boldsymbol{C}^*\right]} \tag{123}$$

where $Tr\left[\boldsymbol{B}_M^*\right] = Tr\left[\boldsymbol{T}_C^P \boldsymbol{B}_M^* \boldsymbol{T}_P^C\right]$ since $\boldsymbol{T}_C^P$ is an orthonormal matrix.

With the α scalar known, we may rewrite the Eq. (122) as

$$\alpha \boldsymbol{C}^* \boldsymbol{T}_C^P - \boldsymbol{T}_C^P \boldsymbol{B}_M^* = \boldsymbol{0}_3 \tag{124}$$

While this equation is exact in the absence of measurement noise or modeling error, the right-hand side will not be exactly zero in practice. Therefore, Modenini suggests finding the attitude $\boldsymbol{T}_C^P$ that minimizes the following cost function

$$\min_{\boldsymbol{T}_C^P \in SO(3)} \quad J(\boldsymbol{T}_C^P) = \|\alpha \boldsymbol{C}^* \boldsymbol{T}_C^P - \boldsymbol{T}_C^P \boldsymbol{B}_M^*\|_F^2 \tag{125}$$

where $\|\cdot\|_F$ indicates the Frobenius norm. Modenini calls Eq. (125) a "modified orthogonal Procrustes problem," and suggests a solution based on the eigendecomposition of $\alpha \boldsymbol{C}^*$ and $\boldsymbol{B}_M^*$. This solution has long been known (see Sec. IV-A3) and takes the form,

$$\boldsymbol{T}_C^P = \boldsymbol{V} \boldsymbol{P} \boldsymbol{W}^T \tag{126}$$

where $\boldsymbol{V}$ and $\boldsymbol{W}$ are obtained from the eigendecomposition of the symmetric matrices $\alpha \boldsymbol{C}^*$ and $\boldsymbol{B}_M^*$,

$$\alpha \boldsymbol{C}^* = \boldsymbol{V} \boldsymbol{D}_C^* \boldsymbol{V}^T \tag{127}$$

$$\boldsymbol{B}_M^* = \boldsymbol{W} \boldsymbol{D}_B^* \boldsymbol{W}^T \tag{128}$$

and where

$$\boldsymbol{P} = \begin{bmatrix} \pm 1 & 0 & 0 \\ 0 & \pm 1 & 0 \\ 0 & 0 & \pm 1 \end{bmatrix} \tag{129}$$

Here, we note that $\boldsymbol{D}_C^* = \boldsymbol{D}_B^*$ are diagonal matrices of eigenvalues, while $\boldsymbol{V}$ and $\boldsymbol{W}$ are matrices with the corresponding unit eigenvectors as columns. The matrix $\boldsymbol{P}$ captures the fact that each of the unit eigenvectors in $\boldsymbol{V}$ and $\boldsymbol{W}$ have ambiguous sign.

The solution in Eq. (126) assumes that the eigenvalues of $\alpha \boldsymbol{C}^*$ and $\boldsymbol{B}_M^*$ (which are the same in the absence of noise) are sorted in a consistent manner, such that their eigenvectors forming the columns of $\boldsymbol{V}$ and $\boldsymbol{W}$ are also sorted in a consistent manner. Any rigorous convention will work. We will revisit this briefly at the end of Section IV-A4.

There are only four unique solutions to Eq. (126) that satisfy the constraint that $\det[\boldsymbol{T}_C^P] = +1$, with only two of these four solutions placing the celestial body in front of the camera. Thus, in general, the horizon fit of a spheroid or triaxial ellipsoid can be used to estimate the spacecraft attitude with a twofold ambiguity.

3) TWO-SIDED ORTHOGONAL PROCRUSTES PROBLEMS

The spacecraft navigator is most likely familiar with the classical (one-sided) orthogonal Procrustes problem within the context of attitude determination from corresponding unit vectors [33], where it is one of the well-known solutions to Wahba's problem [96], [97]. The one-sided orthogonal Procrustes problem takes the form $\min_{\boldsymbol{T}} \|\boldsymbol{A}-\boldsymbol{T}\boldsymbol{B}\|_F^2$ subject to $\boldsymbol{T}^T\boldsymbol{T}=\boldsymbol{I}$, and was studied as early as the 1950s [98]. The modern approach for solving this problem uses the simpler and more robust method developed by Schönemann in the mid-1960s [99].

There are other types of Procrustes problems [100] and we recognize Modenini's cost function from Eq. (125) as the symmetric case of a two-sided orthogonal Procrustes problem,

$$\min_{\boldsymbol{T}\in SO(3)} \|\boldsymbol{T}^T\boldsymbol{A}\boldsymbol{T}-\boldsymbol{B}\|_F^2 \quad \text{or} \quad \min_{\boldsymbol{T}\in SO(3)} \|\boldsymbol{A}\boldsymbol{T}-\boldsymbol{T}\boldsymbol{B}\|_F^2 \tag{130}$$

which also has an elegant solution developed by Schönemann [101] (published in 1968, just two years after his solution to the one-sided orthogonal Procrustes problem). Schönemann's solutions (both one-sided and two-sided) are now well-known and have been studied in great depth [100]. While Modenini (in [95]) is to be commended for independently developing the solution to this problem, claims that this solution is new are inaccurate. The solution procedures of Schönemann and Modenini are identical, with Schönemann's solution predating Modenini's by 50 years.

To reduce confusion and to remain consistent with the larger body of work on Procrustes problems [100], [101], the space navigation community should abandon the name *modified orthogonal Procrustes problem* when referring to this solution. We should instead use the more commonly accepted *two-sided orthogonal Procrustes problem*.

4) OBSERVATIONS ON MODENINI's SOLUTION

A few observations provide some minor efficiency improvements to Modenini's attitude determination solution from [95].

The first observation is on the computation of α. While Modenini leaves this in matrix form, we note the numerator may be rewritten explicitly to obtain

$$\alpha = \frac{a^2+b^2+c^2-r^2}{Tr\left[\boldsymbol{C}^*\right]} \tag{131}$$

where, as before, $r=\|\boldsymbol{r}_C\|=\|\boldsymbol{r}_P\|$.

More importantly, however, we observe that the computation of α is unnecessary and it may be entirely discarded from the solution procedure. To see this explicitly, note that

$$\alpha\boldsymbol{C}^* = \boldsymbol{V}\boldsymbol{D}_C^*\boldsymbol{V}^T \;\rightarrow\; \boldsymbol{C}^* = \boldsymbol{V}(\alpha^{-1}\boldsymbol{D}_C^*)\boldsymbol{V}^T \tag{132}$$

such that the eigenvectors $\boldsymbol{V}$ can be computed by the eigendecomposition of $\boldsymbol{C}^*$ rather than $\alpha\boldsymbol{C}^*$. Since the scalar α doesn't appear anywhere else in the solution procedure, there is no reason to ever compute it in practice. However, we must be careful with the sign ambiguity of $\boldsymbol{C}^*$ when bypassing the computation of α. Since $\boldsymbol{C}^*$ has arbitrary scale and sign, we can choose it such that $\text{sign}(\det[\boldsymbol{C}^*]) = \text{sign}(\det[\boldsymbol{B}_M^*])$. Then we can simply sort the eigenvalues in ascending or descending order, and arrange the eigenvectors in $\boldsymbol{V}$ and $\boldsymbol{W}$ in the same way.

Finally, we sometimes find it convenient to replace $\boldsymbol{V}$ and $\boldsymbol{W}$ with the orthonormal matrices from a singular value decomposition (SVD). In addition to guaranteeing that we use orthonormal matrices in Eq. (126) by construction, the typical SVD algorithm also automatically sorts the singular values (which are the absolute value of the eigenvaleus in this case) from largest to smallest. This ensures the consistent ordering we need here without an explicit check on the sign of $\boldsymbol{C}^*$. The reader is reminded that while $\boldsymbol{C}^*$ and $\boldsymbol{B}_M^*$ are real symmetric matrices, the SVD and eigendecomposition are not the same because the eigenvalues are of mixed sign (i.e., $\boldsymbol{C}^*$ and $\boldsymbol{B}_M^*$ are indefinite matrices). However, the columns in the orthonormal SVD matrices differ from their corresponding unit eigenvector only by factor of ±1, which may be absorbed into the matrix $\boldsymbol{P}$ in Eq. (126). Hence, we may exchange $\boldsymbol{V}$ and $\boldsymbol{W}$ with the orthonormal matrices obtained from an SVD and obtain the same solution for $\boldsymbol{T}_C^P$.

To make this explicit, we may compute the SVD of $\boldsymbol{C}^*$ and $\boldsymbol{B}_M^*$ as

$$\boldsymbol{C}^* = \boldsymbol{V}_L\boldsymbol{S}_C^*\boldsymbol{V}_R^T \tag{133}$$

$$\boldsymbol{B}_M^* = \boldsymbol{W}_L\boldsymbol{S}_B^*\boldsymbol{W}_R^T \tag{134}$$

where $\boldsymbol{V}_L \neq \boldsymbol{V}_R$ and $\boldsymbol{W}_L \neq \boldsymbol{W}_R$. However, the difference in these orthonormal matrices is just a sign change in one (or more) of the columns, which we may also relate to the sorted unit eigenvector matrices,

$$\boldsymbol{V} = \boldsymbol{V}_L\boldsymbol{P}_{VL} = \boldsymbol{V}_R\boldsymbol{P}_{VR} \tag{135}$$

$$\boldsymbol{W} = \boldsymbol{W}_L\boldsymbol{P}_{WL} = \boldsymbol{W}_R\boldsymbol{P}_{WR} \tag{136}$$

where $\boldsymbol{P}_{(\cdot)}$ represents a specific matrix of the form of Eq. (129) that performs the appropriate column-by-column sign changes. Substituting these results into Eq. (126), we see that the SVD orthonormal matrices $\boldsymbol{V}_{(\cdot)}$ and $\boldsymbol{W}_{(\cdot)}$ may replace the eigenvector matrices $\boldsymbol{V}$ and $\boldsymbol{W}$ to obtain

$$\boldsymbol{T}_C^P = \boldsymbol{V}_L\boldsymbol{P}\boldsymbol{W}_L^T \tag{137}$$

where we have made use of the fact that

$$\boldsymbol{P}_{VL}\boldsymbol{P}\boldsymbol{P}_{WL}^T = \begin{bmatrix} \pm1 & 0 & 0 \\ 0 & \pm1 & 0 \\ 0 & 0 & \pm1 \end{bmatrix} \tag{138}$$

In Eq. (137), note that either $\boldsymbol{V}_L$ or $\boldsymbol{V}_R$ may be used on the left, while either $\boldsymbol{W}_L$ or $\boldsymbol{W}_R$ may be used on the right. Regardless of these choices, we obtain the same eight solutions for $\boldsymbol{T}_C^P$ as before (only four of which are right-handed).

5) GENERIC ATTITUDE DETERMINATION WITH THE CONIC LOCUS

We observe that Modenini's solution (see [95] and Sec. IV-A2) operates on the conic envelope instead of the

conic locus. An equivalent solution may be produced using the conic locus, which is just as compact and easy to compute. This is now shown.

Recall that the conic envelope solution started with Eq. (120). To arrive at the conic locus solution, we simply invert this relation [see Eq. (62)] to arrive at the familiar relation from Eq. (59)

$$\boldsymbol{C} \propto \boldsymbol{A}_C \boldsymbol{r}_C \boldsymbol{r}_C^T \boldsymbol{A}_C - (\boldsymbol{r}_C^T \boldsymbol{A}_C \boldsymbol{r}_C - 1)\boldsymbol{A}_C \tag{139}$$

As with the conic envelope solution, rewrite the right-hand side in the celestial body's principal axis frame

$$\boldsymbol{C} \propto \boldsymbol{T}_C^P \left[\boldsymbol{A}_P \boldsymbol{r}_P \boldsymbol{r}_P^T \boldsymbol{A}_P - (\boldsymbol{r}_P^T \boldsymbol{A}_P \boldsymbol{r}_P - 1)\boldsymbol{A}_P \right] \boldsymbol{T}_P^C \tag{140}$$

which, letting $\boldsymbol{M}_P$ be

$$\boldsymbol{M}_P = \boldsymbol{A}_P \boldsymbol{r}_P \boldsymbol{r}_P^T \boldsymbol{A}_P - (\boldsymbol{r}_P^T \boldsymbol{A}_P \boldsymbol{r}_P - 1)\boldsymbol{A}_P \tag{141}$$

simplifies to

$$\boldsymbol{C} \propto \boldsymbol{T}_C^P \boldsymbol{M}_P \boldsymbol{T}_P^C \tag{142}$$

The proportionality relation may be eliminated by introducing the scalar β to obtain

$$\beta \boldsymbol{C} = \boldsymbol{T}_C^P \boldsymbol{M}_P \boldsymbol{T}_P^C \tag{143}$$

which is identical in structure to the conic envelope expression from Eq. (122).

Therefore, following the same procedure as before, we arrive at a two-sided orthogonal Procrustes problem

$$\min_{\boldsymbol{T}_C^P \in SO(3)} \quad J(\boldsymbol{T}_C^P) = \|\beta \boldsymbol{C} \boldsymbol{T}_C^P - \boldsymbol{T}_C^P \boldsymbol{M}_P\|_F^2 \tag{144}$$

We know that the scaling β does not matter and directly compute the solution as

$$\boldsymbol{T}_C^P = \boldsymbol{V}_L \boldsymbol{P} \boldsymbol{W}_L^T \tag{145}$$

where $\boldsymbol{V}_L$ and $\boldsymbol{W}_L$ are obtained form the SVD of $\boldsymbol{C}$ and $\boldsymbol{M}_P$

$$\boldsymbol{C} = \boldsymbol{V}_L \boldsymbol{S}_C \boldsymbol{V}_R^T \tag{146}$$
$$\boldsymbol{M}_P = \boldsymbol{W}_L \boldsymbol{S}_M \boldsymbol{W}_R^T \tag{147}$$

and where $\boldsymbol{P}$ is from Eq. (129). As with the conic envelope solution, we may also write this as an eigendecomposition. From this point forward the conic envelope and conic locus solutions are identical.

The similarities between the conic locus and conic envelope solutions should come as no surprise, since taking the inverse of a symmetric matrix doesn't change its eigenvectors. It is easy to see that the orthonormal matrix of eigenvectors $\boldsymbol{U}$ for a $n \times n$ symmetric matrix $\boldsymbol{A}$ (no relation to planet shape matrix) is the same as for the matrix $\boldsymbol{A}^{-1}$

$$\boldsymbol{A} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{U}^T \;\rightarrow\; \boldsymbol{A}^{-1} = \boldsymbol{U}\boldsymbol{D}^{-1}\boldsymbol{U}^T \tag{148}$$

where $\boldsymbol{U}\boldsymbol{U}^T = \boldsymbol{I}_n$ and $\boldsymbol{U}^{-1} = \boldsymbol{U}^T$.

6) COMPARISON OF CONIC ENVELOPE AND CONIC LOCUS ALGORITHMS

The conic envelope and conic locus solutions will always produce the exact same numerical answer, so there is no benefit of one over the other with regards to accuracy. The only difference lies in the number of operations required on a digital computer. To count the number of operations, we begin with the assumption that the conic fitting algorithms provide an estimate of the conic locus (i.e., they provide $\boldsymbol{C}$ and not $\boldsymbol{C}^*$), as is the case for every algorithm discussed in Sec. II-H. We also assume that $\boldsymbol{A}_P$ and $\boldsymbol{A}_P^{-1}$ may be precomputed. Under these circumstances, observe that the two solutions are computationally identical once we have $\boldsymbol{C}$ and $\boldsymbol{M}_P$ for the conic locus or $\boldsymbol{C}^*$ and $\boldsymbol{B}_M^*$ for the conic envelope.

For the case of the conic locus, $\boldsymbol{C}$ is provided directly from the conic fitting algorithm and we only need to compute $\boldsymbol{M}_P$. Taking advantage of matrix symmetry and that $\boldsymbol{A}_P$ is diagonal, we may compute $\boldsymbol{M}_P$ with 15 multiplies and 6 adds. Conversely, for the conic envelope, we must compute both $\boldsymbol{C}^*$ and $\boldsymbol{B}_M^*$. Taking advantage of the same matrix properties as in the conic locus case, we may compute $\boldsymbol{C}^*$ and $\boldsymbol{B}_M^*$ with 18 multiplies and 9 adds. Therefore, we see that the conic locus solution provides the exact same answer as the conic envelope solution, but achieves this result with fewer operations on a digital computer.

We conclude by remarking that the computational advantages of the conic locus approach are small and that the two methods are essentially equivalent in practice. This is the case since the computational costs described above are negligible when compared to the cost of performing the ellipse fit to find $\boldsymbol{C}$. The primary purpose of this discussion is to dispel the erroneous notion that the conic envelope solution is simpler or more computationally efficient than the conic locus solution—when, in fact, the reverse is true for horizon-based attitude determination. Thus, there is no especially meaningful argument to use one over the other and the reader should feel free to choose whichever method they prefer.

7) ATTITUDE OBSERVABILITY

If the observed celestial body is a sphere, it was discussed in Sec. IV-A1 that only two components of attitude are observable. Issues of observability in this situation are straightforward and were fully discussed in Sec. IV-A1. More complicated is the issue of attitude observability when imaging a spheroid or triaxial ellipsoid, and this is the situation that is now discussed.

Regardless of the representation of the horizon (conic locus or conic envelope), the two-sided orthogonal Procrustes problem presents a fourfold ambiguity. Specifically, in either Eq. (126), Eq. (137) or Eq. (145), there are four choices of $\boldsymbol{P}$ that provide a proper orthogonal matrix with $\det(\boldsymbol{T}_C^P) = +1$. Ensuring that $\boldsymbol{T}_C^P$ is a proper orthogonal matrix may be achieved in a variety of ways. One such way is to define

$$\boldsymbol{P}_1 = \text{diag}[1, 1, 1] \tag{149a}$$
$$\boldsymbol{P}_2 = \text{diag}[-1, 1, 1] \tag{149b}$$

$$\boldsymbol{P}_3 = \text{diag}[1, -1, 1] \tag{149c}$$
$$\boldsymbol{P}_4 = \text{diag}[1, 1, -1] \tag{149d}$$

and then compute the corresponding proper orthogonal matrix by substituting these possibilities for $\boldsymbol{P}_i$ into Eq. (126)

$$\{\boldsymbol{T}_C^P\}_i = \det[\boldsymbol{V}\boldsymbol{P}_i\boldsymbol{W}^T]\,\boldsymbol{V}\boldsymbol{P}_i\boldsymbol{W}^T, \quad i = 1, 2, 3, 4 \tag{150}$$

All four of these solutions satisfy the original two-sided orthogonal Procrustes problem.

Only two of the four solutions in Eq. (150) will place the celestial body in front of the camera. The process of checking to ensure that an observed object lies in front of (and not behind) the camera is sometimes called a *cheirality test* [45], [102]. To perform such a cheirality test, we simply need to see if the z-component of $\boldsymbol{r}_C$ is positive, which occurs when

$$\boldsymbol{k}^T\boldsymbol{r}_C = \boldsymbol{k}^T\boldsymbol{T}_C^P\boldsymbol{r}_P > 0 \tag{151}$$

where $\boldsymbol{k}^T = [0, 0, 1]$. This results in a twofold ambiguity in the relative attitude $\boldsymbol{T}_C^P$.

It is impossible to remove this twofold ambiguity using the horizon fit alone. Of course, disambiguation is possible when other information exists. For example, if the direction to the Sun is known in the camera frame (e.g., from a Sun sensor), this could be used to remove the ambiguity. Other forms of additional attitude information may also remove this ambiguity.

8) ALGORITHM SUMMARY AND PERFORMANCE COMPARISON

The conic envelope (see Sec. IV-A2) and conic locus (see Sec. IV-A5) solutions provide identical attitude estimates. As an example, Algorithm 2 shows pseudocode for the conic locus method since this method is marginally more efficient than the conic envelope method (see Sec. IV-A6). Regardless, the implementation of these techniques is very similar.

Algorithm 2 Pseudocode for attitude determination algorithm with horizon fit of a spheroid or triaxial ellipsoid. Assume relative position is known in the celestial body's frame

procedure $\{\boldsymbol{T}_C^P\}_{j=1}^2 = \text{AttDetP}(\boldsymbol{C}, \boldsymbol{r}_P, a, b, c)$
  compute $\boldsymbol{A}_P = \text{diag}[1/a^2, 1/b^2, 1/c^2]$ ▷ Eq. (24)
  compute $\boldsymbol{M}_P$ ▷ Eq. (141)
  $[\boldsymbol{V}_L, \boldsymbol{S}_C, \boldsymbol{V}_R] = \text{svd}(\boldsymbol{C})$ ▷ Eq. (146)
  $[\boldsymbol{W}_L, \boldsymbol{S}_M, \boldsymbol{W}_R] = \text{svd}(\boldsymbol{M}_P)$ ▷ Eq. (147)
  $j = 1$
  for $i = 1$ to 4 **do**
    compute $\boldsymbol{P}_i$ ▷ Eq. (149)
    compute $\boldsymbol{T} = \boldsymbol{V}_L\boldsymbol{P}_i\boldsymbol{W}_L^T$
    compute $\boldsymbol{T} = \det[\boldsymbol{T}]\,\boldsymbol{T}$ ▷ Eq. (150)
    if $\boldsymbol{k}^T\boldsymbol{T}\boldsymbol{r}_P > 0$ **then**
      record $\{\boldsymbol{T}_C^P\}_j = \boldsymbol{T}$
      $j = j + 1$
return $\{\boldsymbol{T}_C^P\}_{j=1}^2$

As an example, consider an Earth-orbiting spacecraft at a range of 45,000 km from the planet center (just beyond GEO). Assume that we know the relative position $\boldsymbol{r}_P$, but do not know the relative attitude $\boldsymbol{T}_C^P$. Earth is used for this example because Earth is the only celestial body where such a scenario is expected to occur in practice. While considering celestial bodies other than Earth in this scenario may be of academic interest, such numerical results are of little practical relevance.

Suppose we image the Earth with a LWIR thermal camera. In this example, we simulate a camera with a 20 deg FOV and a 640×640 pixel FPA. Such an imaging system is in-family with commercially available LWIR cameras (circa 2020).

Let the Earth be modeled as an oblate spheroid. Since observations are done in the infrared, we inflate the dimensions of the reference ellipsoid by 40 km to account for the apparent altitude of the CO_2 band (at wavelengths of 14.0–16.3 μm) as discussed in Sec. II-G. This produces an Earth ellipsoid with an equatorial radius of $a = b = 6,418.1$ km and a polar radius of $c = 6,396.8$ km. In this example, the Earth projects to an image ellipse with semimajor axis of 261.81 pixels and semiminor axis of 261.05 pixels. We assume that the method from [53] is used for horizon localization, producing image processing errors of about $\sigma = 0.07$ pixel.

The attitude determination performance for this example scenario is summarized in Table 4 and Table 5. Attitude determination errors are found to be about 2.6 deg for all of the ellipse fitting algorithms considered, once again suggesting that the non-iterative semihyper least-squares method from [60] is likely the most suitable for onboard navigation. We observe the mean to be small compared to the standard deviation. We also observe that nearly all of the attitude error is about the camera boresight (z-axis) direction for reasons that are discussed later (see Sec. IV-C). When interpreting the results from Table 4 and Table 5, the reader is reminded that different vantage points and different camera parameters can produce drastically different performance.

To better highlight how performance changes from different vantage points, suppose we have the same example LWIR camera as before. Now, consider the special case where the

TABLE 4. Attitude determination (with known r_P) performance statistics for synthetic images of the Earth in the LWIR. Attitude determination performed using Algorithm 2. Statistics are computed from a 10,000-run Monte Carlo simulation, with each algorithm operating on the same set of simulation inputs.

Method	Mean Error (deg)	Standard Deviation (deg)
Non-Iterative Ellipse Methods		
Modified Fitzgibbon [62], [63]	0.5977	2.6790
Semihyper Least-Squares [60]	0.1875	2.6031
Hyper Least-Squares [60]	0.1875	2.6031
Iterative Ellipse Methods		
Doubly-Optimal Fit [66]	0.1874	2.6030
Fund. Numerical Scheme [65]	0.1877	2.6031
Szpak, et al. [67]	0.1912	2.6037

TABLE 5. Attitude determination (with known r_P) covariance for synthetic images of the Earth in the LWIR. The root-sum-square (RSS) of these values are equivalent to the right-most column in Table 4. Errors are expressed in the camera frame.

Method	σ_x (deg)	σ_y (deg)	σ_z (deg)
Non-Iterative Ellipse Methods			
Modified Fitzgibbon [62], [63]	0.0162	0.0920	2.6774
Semihyper Least-Squares [60]	0.0156	0.0894	2.6015
Hyper Least-Squares [60]	0.0156	0.0894	2.6015
Iterative Ellipse Methods			
Doubly-Optimal Fit [66]	0.0156	0.0894	2.6014
Fund. Numerical Scheme [65]	0.0156	0.0894	2.6015
Szpak, et al. [67]	0.0156	0.0894	2.6021

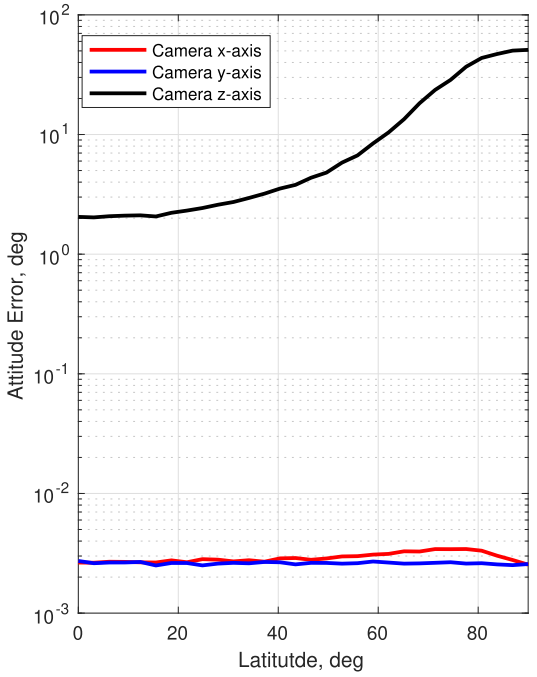

FIGURE 17. Standard deviation of attitude error for synthetic images of Earth in the LWIR. Statistics at each different latitude are computed from a 2,000-run Monte Carlo simulation. Attitude determination performed using Algorithm 2.

camera is perfectly nadir pointed (i.e., $\boldsymbol{r}_C^T = [0, 0, r]$) and that $r = 45,000$ km. Let the position in the Earth's principal axis frame vary from a latitude of 0 deg (in the equatorial plane) to a latitude of 90 deg (above the north pole). The resulting attitude determination errors expressed in the camera frame are shown in Fig. 17. We see that the cross-boresight attitude errors (red and blue lines) are approximately constant for all latitudes and remain small. These errors remain small and constant in this example because (1) our ability to discern the direction to Earth's center remains stable for all latitudes and (2) the Earth's center is positioned exactly along the boresight direction (hence, there is no projection of attitude errors about the direction $\boldsymbol{r}_C$ onto the x-axis and y-axis directions). At low latitudes, we are able to compute the camera roll about the boresight (direction of $\boldsymbol{r}_C$) to within a few degrees using the oblateness of the Earth. This is similar to what was observed under different conditions in Table 5. At high latitudes the Earth's oblateness becomes less apparent, with the Earth's oblateness becoming completely unobservable at $\phi = 90$ deg (i.e., when the camera center lies on the oblate spheroid's axis of rotational symmetry). As the Earth's oblateness becomes less observable, our ability to use the oblateness to infer roll about the camera boresight direction deteriorates (black line in Fig. 17)—with complete loss of one degree of attitude observability at $\phi = 90$ deg. In this particular example, we exceed an error of $\sigma_z \geq 10$ deg when the spacecraft position is at a latitude above ~ 60 deg.

B. FINDING CELESTIAL BODY ATTITUDE (POSITION KNOWN IN THE CAMERA FRAME, C)

The second attitude determination scenario of interest is when the the relative position is known in the spacecraft camera frame (known $\boldsymbol{r}_C$). As discussed earlier, this situation could arise during interplanetary exploration when visiting a celestial body other than Earth.

1) ATTITUDE DETERMINATION WITH CONIC ENVELOPE

A straightforward solution exists by making use of the conic envelope. Therefore, proceed by first first recalling Eq. (120)

$$\alpha \boldsymbol{C}^* = \boldsymbol{A}_C^{-1} - \boldsymbol{r}_C \boldsymbol{r}_C^T \tag{152}$$

Since it is possible to compute α from Eq. (131), we may rearrange to find

$$\boldsymbol{A}_C^{-1} = \alpha \boldsymbol{C}^* + \boldsymbol{r}_C \boldsymbol{r}_C^T \tag{153}$$

Define the symmetric matrix $\boldsymbol{G}$ as

$$\boldsymbol{G} = \alpha \boldsymbol{C}^* + \boldsymbol{r}_C \boldsymbol{r}_C^T \tag{154}$$

Therefore, substituting for $\boldsymbol{A}_C$ from Eq. (28), the original conic envelope equation becomes

$$\boldsymbol{T}_C^P \boldsymbol{A}_P^{-1} \boldsymbol{T}_P^C = \alpha \boldsymbol{C}^* + \boldsymbol{r}_C \boldsymbol{r}_C^T = \boldsymbol{G} \tag{155}$$

Now, to make things explicit, rewrite this as

$$\boldsymbol{G} = \boldsymbol{T}_C^P \boldsymbol{A}_P^{-1} \boldsymbol{T}_P^C \tag{156}$$

which we observe to be of identical form as Eq. (122) and Eq. (143). Thus, we again arrive at a two-sided orthogonal Procrustes problem,

$$\min_{\boldsymbol{T}_C^P \in SO(3)} \quad J(\boldsymbol{T}_C^P) = \|\boldsymbol{G}\boldsymbol{T}_C^P - \boldsymbol{T}_C^P \boldsymbol{A}_P^{-1}\|_F^2 \tag{157}$$

Following the exact same procedure as before, we directly compute the solution as

$$\boldsymbol{T}_C^P = \boldsymbol{V}\boldsymbol{P}\boldsymbol{W}^T \tag{158}$$

where $\boldsymbol{P}$ is from Eq. (129) and where $\boldsymbol{V}$ and $\boldsymbol{W}$ are obtained from the SVD of $\boldsymbol{G}$ and $\boldsymbol{A}_P^{-1}$

$$\boldsymbol{G} = \boldsymbol{V}\boldsymbol{S}_G\boldsymbol{V}^T \tag{159}$$

$$\boldsymbol{A}_P^{-1} = \boldsymbol{W}\boldsymbol{S}_A\boldsymbol{W}^T \tag{160}$$

In this case, the SVD is the same as the eigendecomposition since $\boldsymbol{G}$ and $\boldsymbol{A}_P^{-1}$ are symmetric positive definite. Observe that $\boldsymbol{A}_P^{-1}$ is diagonal by construction [see Eq. (24) for $\boldsymbol{A}_P$],

$$\boldsymbol{A}_P^{-1} = \text{diag}[a^2, b^2, c^2] \tag{161}$$

Thus, the matrix $\boldsymbol{W}$ is a permutation matrix when a, b, c are unique. Under the assumption that the singular values in $\boldsymbol{S}_A$ are arranged in descending order, $\boldsymbol{W}$ does nothing more than rearrange the diagonals of $\boldsymbol{A}_P^{-1}$ to also be in descending order. In the special case where $a > b > c$, we find that

$$\boldsymbol{W} = \boldsymbol{I}_3 \tag{162}$$

though other values for $\boldsymbol{W}$ are possible for spheres and oblate spheroids or if a different ordering of a, b, c is chosen.

2) ATTITUDE OBSERVABILITY

The attitude observability is poorer when the relative position is known in the camera frame ($\boldsymbol{r}_C$) than when the relative position is known in the celestial body's frame ($\boldsymbol{r}_P$). This may be seen by comparing the results below with those of Sec. IV-A7.

In the case of a sphere, the relative attitude is completely unobservable when the relative position is known in the camera frame. This matches simple intuition, as a sphere at a given $\boldsymbol{r}_C$ may be arbitrarily rotated with no change in its horizon's appearance in an image.

In the case of a spheroid, we may only determine the direction of the axis of rotational symmetry. Again, this matches intuition, as a spheroid at position $\boldsymbol{r}_C$ may be arbitrarily rotated about its axis of symmetry with no change in the horizon's appearance in an image. The most common case is an oblate spheroid ($a = b > c$) and we assume without loss of generality that the axis of symmetry is coincident with the z-axis of the celestial body's principal axis frame (thus, the body has a principal axis length of c along the z-direction). Denote the axis of symmetry by the unit vector $\boldsymbol{w}$. Thus, by construction, the axis of symmetry as expressed in the body's principal axis frame $\boldsymbol{w}_P$ is

$$\boldsymbol{w}_P = \pm\boldsymbol{k} = \begin{bmatrix} 0 \\ 0 \\ \pm 1 \end{bmatrix} \tag{163}$$

The $\pm$ is used to explicitly highlight that the direction of the axis of symmetry is unobservable. To express the axis of symmetry in the camera frame instead of the celestial body's principal axis frame is trivial,

$$\boldsymbol{w}_C = \boldsymbol{T}_C^P \boldsymbol{w}_P = \pm\boldsymbol{T}_C^P \boldsymbol{k} \tag{164}$$

and, therefore,

$$\boldsymbol{w}_C = \pm\boldsymbol{T}_C^P \boldsymbol{k} = \pm\boldsymbol{V}\boldsymbol{P}\boldsymbol{W}^T\boldsymbol{k} \tag{165}$$

When using the convention that $a = b > c$ with c along the z-axis direction, we know that the smallest singular value of $\boldsymbol{A}_P^{-1}$ is unique and corresponds to the z-direction—thus constraining the structure of $\boldsymbol{W}$. Taking this into consideration and substituting for $\boldsymbol{W}$ and $\boldsymbol{k}$ in Eq. (165), the solution for the $\boldsymbol{w}_C$ becomes

$$\boldsymbol{w}_C = \pm\boldsymbol{V}\begin{bmatrix} \cdot & \cdot & 0 \\ \cdot & \cdot & 0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \pm\boldsymbol{V}\boldsymbol{k} \tag{166}$$

Thus, partitioning $\boldsymbol{V}$ by column,

$$\boldsymbol{V} = \begin{bmatrix} \boldsymbol{v}_1 & \boldsymbol{v}_2 & \boldsymbol{v}_3 \end{bmatrix} \tag{167}$$

we arrive at the solution

$$\boldsymbol{w}_C = \pm\boldsymbol{v}_3 \tag{168}$$

This strikingly simple relation states that the celestial body's axis of symmetry (for an oblate spheroid) as expressed in the camera frame is along the direction described by the last column of $\boldsymbol{V}$. The reader is reminded that $\boldsymbol{V}$ comes form the SVD of $\boldsymbol{G}$ [see Eq. (159)]. Thus, we note that all of the observable attitude information may be computed directly from the SVD of $\boldsymbol{G}$, without the need to ever compute $\boldsymbol{W}$.

In the case of a triaxial ellipsoid, we may determine the complete attitude up to a fourfold ambiguity. Specifically, due to possible values of $\boldsymbol{P}$ from Eq. (129), the solution from Eq. (158) produces four unique attitudes with $\det[\boldsymbol{T}_C^P] = +1$. It also produces four results with $\det[\boldsymbol{T}_C^P] = -1$, but these are discarded since they describe left-handed coordinate systems. Thus using the conventions leading to $\boldsymbol{W} = \boldsymbol{I}_3$ from Eq. (162), the four solutions are

$$\{\boldsymbol{T}_C^P\}_i = \det[\boldsymbol{V}\boldsymbol{P}_i]\,\boldsymbol{V}\boldsymbol{P}_i \quad i = 1, 2, 3, 4 \tag{169}$$

where $\boldsymbol{P}_i$ is given by Eq. (149). All four of these attitude estimates are equally plausible, and it is impossible to determine which of these are correct from only the apparent horizon in a single image.

3) ALGORITHM SUMMARY AND PERFORMANCE COMPARISON

The developments from the previous subsections result in Algorithm 3 for attitude determination with an oblate spheroid and in Algorithm 4 for attitude determination with a triaxial ellipsoid. These algorithms are simple to implement in only a few lines of code.

Algorithm 3 Pseudocode for attitude determination algorithm with horizon fit of an oblate spheroid. Assume relative position is known in the camera frame

procedure $\boldsymbol{w}_C = \text{AttDetC\_ObSp}(\boldsymbol{C}^*, \boldsymbol{r}_C, a, c)$
  compute $\boldsymbol{A}_P^{-1} = \text{diag}[a^2, a^2, c^2]$ ▷ Eq. (161)
  compute $r = \|\boldsymbol{r}_C\|$
  compute α ▷ Eq. (131)
  compute $\boldsymbol{G} = \alpha\boldsymbol{C}^* + \boldsymbol{r}_C\boldsymbol{r}_C^T$ ▷ Eq. (154)
  $[\boldsymbol{V}, \boldsymbol{S}_G, \boldsymbol{V}] = \text{svd}(\boldsymbol{G})$ ▷ Eq. (159)
  compute $\boldsymbol{w}_C = \boldsymbol{v}_3$ ▷ Eq. (168)
return $\boldsymbol{w}_C$

Solving an attitude determination problem when the relative position is known in the camera frame (known $\boldsymbol{r}_C$) makes the most sense within the context of interplanetary exploration. As an example, consider the dwarf planet Ceres. We may model Ceres as an oblate spheroid with an equatorial radius of $a = 482.1$ km and a polar radius of $c = 445.9$ km. Ceres is chosen here because it is an oblate spheroid with no

Algorithm 4 Pseudocode for attitude determination algorithm with horizon fit of a triaxial ellipspod. Assume relative position is known in the camera frame

procedure $\{\boldsymbol{T}_C^P\}_{i=1}^4 = \text{AttDetC\_El}(\boldsymbol{C}^*, \boldsymbol{r}_C, a, b, c)$
  compute $\boldsymbol{A}_P^{-1} = \text{diag}[a^2, b^2, c^2]$ ▷ Eq. (161)
  compute $r = \|\boldsymbol{r}_C\|$
  compute α ▷ Eq. (131)
  compute $\boldsymbol{G} = \alpha\boldsymbol{C}^* + \boldsymbol{r}_C\boldsymbol{r}_C^T$ ▷ Eq. (154)
  $[\boldsymbol{V}, \boldsymbol{S}_G, \boldsymbol{V}] = \text{svd}(\boldsymbol{G})$ ▷ Eq. (159)
  for $i = 1$ to 4 **do**
    compute $\boldsymbol{P}_i$ ▷ Eq. (149)
    compute $\{\boldsymbol{T}_C^P\}_i = \det[\boldsymbol{V}\boldsymbol{P}_i]\,\boldsymbol{V}\boldsymbol{P}_i$ ▷ Eq. (169)
return $\{\boldsymbol{T}_C^P\}_{i=1}^4$

atmosphere and because it has been used as the case study in many recent papers on this topic (e.g., Ref. [95]).

Since Ceres is an airless body, we may use a high-resolution camera that operates in the visible spectrum. Therefore, assume a similar imaging system as from the OPNAV example: 20 deg FOV camera, $2,048 \times 2,048$ pixel FPA, and horizon localization with $\sigma = 0.07$ pixel. Under the conditions in this example, Ceres projects to an image ellipse with semimajor axis of 285.82 pixels and semiminor axis of 267.47 pixels.

Attitude determination results from a 10,000-run Monte Carlo analysis are summarized in Table 6. Because Ceres is an oblate spheroid, we may only compute the direction of the axis of symmetry in the camera frame ($\boldsymbol{w}_C$). The results in Table 6 show the statistics for the error in this unit vector direction, which we find to be about 1.6 deg in this particular case. As before, the reader is reminded that different vantage points and different camera parameters can produce drastically different attitude determination performance.

TABLE 6. Attitude determination (with known r_C) performance statistics for synthetic images of the dwarf planet Ceres. Results indicate errors in the direction of the oblate spheroid axis of symmetry, w_C. Attitude determination performed using Algorithm 3. Statistics are computed from a 10,000-run Monte Carlo simulation, with each algorithm operating on the same set of simulation inputs.

Method	Mean Error (deg)	Standard Deviation (deg)
Non-Iterative Ellipse Methods		
Modified Fitzgibbon [62], [63]	0.3450	1.6065
Semihyper Least-Squares [60]	0.1053	1.6086
Hyper Least-Squares [60]	0.1053	1.6086
Iterative Ellipse Methods		
Doubly-Optimal Fit [66]	0.1053	1.6077
Fund. Numerical Scheme [65]	0.1051	1.6077
Szpak, et al. [67]	0.1113	1.6074

C. COMPARISON OF ATTITUDE DETERMINATION PERFORMANCE IN DIFFERENT FRAMES

The distinction about the frame in which the relative position is known (i.e., do we know $\boldsymbol{r}_P$ or $\boldsymbol{r}_C$) may not seem particularly important at first glance. Indeed, many authors have not made clear which of these is known in their analyses and some have even confounded knowledge of the two. We now show the distinction between knowledge of $\boldsymbol{r}_P$ and $\boldsymbol{r}_C$ to be of critical importance and show that these two situations describe fundamentally different geometric problems. This is best accomplished by an example.

Consider attitude determination of Miamis (see Fig. 18). One of the moons of Saturn, Miamis makes an interesting test case as one of the Solar System's smallest ellipsoidal bodies. Using shape models developed from Cassini observations [103], we approximate Mimas as a triaxial ellipsoid with principal axis dimensions $a = 207.8$ km, $b = 196.7$ km, and $c = 190.6$ km.

FIGURE 18. Images of Saturn's moon Miamis collected by the Cassini spacecraft's Imaging Science Subsystem. Raw images are available from [37].

Since Miamis is an airless body, assume the same visible spectrum imaging system as from the OPNAV example: 20 deg FOV camera, $2,048 \times 2,048$ pixel FPA, and horizon localization with $\sigma = 0.07$ pixel. Under the conditions in this example, Miamis projects to an image ellipse with semimajor axis of 162.72 pixels and semiminor axis of 152.29 pixels.

Numerical results from a 10,000-run Monte Carlo analysis are shown in Table 7 for the case when $\boldsymbol{r}_P$ is known. Numerical results from the same Monte Carlo runs are shown in Table 8 for the case when $\boldsymbol{r}_C$ is known.

It is immediately evident that the attitude determination is about two orders of magnitude better when the relative position is known in the celestial body frame P instead of in the camera frame C. Such a large difference may seem counter-intuitive since both problems start with the same initial equation, with the only difference being the frame in which $\boldsymbol{r}$ is known. They key to understanding this it to recognize that the image fundamentally contains information about the direction to the celestial body in the camera frame (i.e., $\boldsymbol{e}_C = \boldsymbol{r}_C/\|\boldsymbol{r}_C\|$), such that knowing $\boldsymbol{r}_P$ from external

TABLE 7. Attitude determination (with known r_P) covariance for synthetic images of the Saturnian moon Miamis. Attitude determination performed using Algorithm 2. Errors are expressed in the camera frame. Statistics are computed from a 10,000-run Monte Carlo simulation, with each algorithm operating on the same set of simulation inputs.

Method	σ_x (deg)	σ_y (deg)	σ_z (deg)
Non-Iterative Ellipse Methods			
Modified Fitzgibbon [62], [63]	0.0204	0.0041	0.1483
Semihyper Least-Squares [60]	0.0203	0.0041	0.1478
Hyper Least-Squares [60]	0.0203	0.0041	0.1478
Iterative Ellipse Methods			
Doubly-Optimal Fit [66]	0.0203	0.0041	0.1478
Fund. Numerical Scheme [65]	0.0203	0.0041	0.1478
Szpak, et al. [67]	0.0203	0.0041	0.1479

TABLE 8. Attitude determination (with known r_C) covariance for synthetic images of the Saturnian moon Miamis. Attitude determination performed using Algorithm 4. Errors are expressed in the camera frame. Statistics are computed from a 10,000-run Monte Carlo simulation, with each algorithm operating on the same set of simulation inputs.

Method	σ_x (deg)	σ_y (deg)	σ_z (deg)
Non-Iterative Ellipse Methods			
Modified Fitzgibbon [62], [63]	8.9139	11.0354	3.3474
Semihyper Least-Squares [60]	8.3296	11.7751	3.1572
Hyper Least-Squares [60]	8.3296	11.7751	3.1572
Iterative Ellipse Methods			
Doubly-Optimal Fit [66]	8.3256	11.7726	3.1551
Fund. Numerical Scheme [65]	8.3261	11.7707	3.1550
Szpak, et al. [67]	8.5539	11.5094	3.2136

sources provides extra attitude information while knowing $\boldsymbol{r}_C$ does not. Moreover, we note that horizon fitting produces an image ellipse that is more sensitive to the *location* of the body in the image than to the *orientation* of the ellipse in the image. This is especially true as the ellipticity of the image ellipse becomes smaller. Therefore, the most observable attribute from a horizon fit loosely corresponds to knowledge of $\boldsymbol{e}_C$.

If $\boldsymbol{r}_P$ is known, then $\boldsymbol{e}_P = \boldsymbol{r}_P/\|\boldsymbol{r}_P\|$ is known. By knowing a unit vector in two different frames (e.g., given $\boldsymbol{e}_P$ and measure $\boldsymbol{e}_C$) we may compute the two components of attitude that take this vector in one frame onto itself in the other frame. There is no attitude information for rotations about the direction of that unit vector. Attitude information about this unit vector direction must come from another source—which, in this case, is the apparent orientation the elliptical horizon in the image.

Consequently, knowing $\boldsymbol{r}_P$ allows us to obtain an excellent estimate of the two components of attitude that rotate $\boldsymbol{r}_P$ onto $\boldsymbol{r}_C$. Rotation about the direction to the celestial body comes from the orientation of the projected horizon, which is much less observable. Thus, recognizing that $\boldsymbol{r}_C$ is mostly along the boresight (camera z-axis) direction for a modest FOV camera, we understand why the attitude errors in Table 7 are much smaller in the camera x-axis and y-axis directions than in the z-axis direction. We remind the reader that such an explicit mechanization of the geometry is not necessary and is implicitly carried out by the more compact mathematics in Algorithm 4.

When we only know $\boldsymbol{r}_C$ (and not $\boldsymbol{r}_P$) from external sources, all components of the attitude must be inferred from the shape of the projected image ellipse. Since this is less observable than the location of the ellipse (especially for bodies that are nearly spherical), we find substantially worse attitude determination performance in these cases. Hence the larger attitude errors in Table 8 (as compared to Table 7).

V. OPTICAL NAVIGATION (OPNAV) WITH A BODY OF UNKNOWN ATTITUDE

The last of the three motivating scenarios from Sec. I is the case where we collect an image of a nearby celestial body with unknown relative position and unknown relative attitude. It is impossible to reconstruct the full six degree-of-freedom relative state from a single image of a celestial body modeled as a proper quadric surface (sphere, spheroid, or triaxial ellipsoid). This fact becomes clear when we recall that the horizon arc for such bodies appears as a conic section in an image—and conics only have five degrees-of-freedom. At most we may recover five components of the relative state, though we recover less than five degrees-of-freedom in many cases due to degeneracies (or near degeneracies) in the celestial body's global shape.

In the case of a spherical celestial body ($a = b = c$), the body's tightly bounding cone is the same for all attitudes. We observe that $\boldsymbol{A}_C = \boldsymbol{A}_P = a^{-2}\boldsymbol{I}_3$ for any attitude $\boldsymbol{T}_C^P$ and conclude that it is impossible extract relative attitude information from just the horizon when the relative position is also unknown. While the relative attitude of a sphere is completely unobservable, we may unambiguously estimate the relative position in the camera frame $\boldsymbol{r}_C$. It is impossible to transform this result into $\boldsymbol{r}_P$ since the attitude of the celestial body remains unknown. The best way to solve for $\boldsymbol{r}_C$ in practice is with the Christian-Robinson algorithm from Sec. III-C.

Many celestial bodies in the Solar System are well-modeled as oblate spheroids (e.g., $a = b > c$). In this case, it is often possible to find a five degree-of-freedom pose solution. Some preliminary work for this case was recently presented by Modenini in [104]. Unfortunately, though theoretically sound and an important contribution, the results of [104] rely on unnecessary geometric constructs and do not consider the frames in which the OPNAV measurements most natively exist. This complicates the solution, obscures otherwise straightforward geometric results, and prevents the algorithm's extension to the case of a triaxial ellipsoid. In this tutorial we present a new way of solving the problem of pose estimation with an oblate spheroid. Here, we suggest that an easier way to represent the pose measurement is with the relative position in the camera frame $\boldsymbol{r}_C$ and the unit vector along the oblate spheroid's axis of revolution direction. Specifying only the direction of the spheroid's axis of revolution provides two degrees-of-freedom in attitude, allowing the body to spin about this axis (a rotation which is clearly unobservable from only the horizon projection). It is impossible to unambiguously transform this result into $\boldsymbol{r}_P$, since the complete attitude of the celestial body is unobservable.

In the case of a sphere and spheroid, the repeated dimension of one (or more) of the celestial body's principal axis lengths isolates the missing degree(s)-of-freedom in pose to just the attitude. This allows us to directly estimate the relative position $\boldsymbol{r}_C$ and, in the case of a spheroid, also estimate the celestial body's axis of revolution. For the case of a generic triaxial ellipsoid ($a \neq b \neq c$), each axis length is unique and the missing degree-of-freedom now contains elements of both the position and the attitude. The possible pose estimates now lie along a one-dimensional manifold in a six-dimensional state space.

The solution for celestial bodies of all quadric shapes begins with the expression for the conic envelope from Eq. (61)

$$\boldsymbol{C}^* \propto \boldsymbol{A}_C^{-1} - \boldsymbol{r}_C \boldsymbol{r}_C^T \tag{170}$$

which we turn into an equality relation by introducing the scaling α,

$$\alpha \boldsymbol{C}^* = \boldsymbol{A}_C^{-1} - \boldsymbol{r}_C \boldsymbol{r}_C^T \tag{171}$$

The reader will recognize this as the same starting point for the conic envelope solution for the horizon-based attitude determination problem [see Eqs. (120) and (152)].

What follows is a detailed discussion of the solution procedure for celestial bodies of all proper quadric shapes. While the discussion is lengthy, the resulting algorithms for pose estimation are simple to implement and are summarized in Section V-E.

A. BOUNDS ON SCALE FACTOR FROM THE RANK-ONE UPDATE OF A DIAGONAL MATRIX

The scale factor α in Eq. (171) cannot be computed via Eq. (131) since the range $r = \|\boldsymbol{r}_P\| = \|\boldsymbol{r}_C\|$ is unknown. A different approach is required here, where we instead seek the bounds of possible values for α. Bounds on the scale factor α may be found using constraints that arise from the known eigenvalues of $\boldsymbol{C}^*$ and $\boldsymbol{A}_P$. One straightforward way to arrive at these bounds is to rewrite Eq. (171) as a rank-one update of a diagonal matrix. This may be achieved by diagonalizing either $\boldsymbol{C}^*$ or $\boldsymbol{A}_P$.

We first consider the constraints arising from the diagonalization of $\boldsymbol{C}^*$. Define $\{\lambda_i\}_{i=1}^3$ to be the three eigenvalues of $\boldsymbol{C}^*$. Now, taking the eigendecomposition of $\alpha \boldsymbol{C}^*$, we find that

$$\alpha \boldsymbol{C}^* = \alpha \boldsymbol{V} \boldsymbol{\Lambda} \boldsymbol{V}^T \tag{172}$$

where the matrices $\boldsymbol{\Lambda}$ and $\boldsymbol{V}$ are

$$\boldsymbol{\Lambda} = \text{diag}[\lambda_1, \lambda_2, \lambda_3] \quad \text{and} \quad \boldsymbol{V} = \begin{bmatrix} \boldsymbol{v}_1 & \boldsymbol{v}_2 & \boldsymbol{v}_3 \end{bmatrix} \tag{173}$$

Since the sign of $\boldsymbol{C}^*$ is arbitrary and one of its eigenvalues has unique sign, choose the sign of $\boldsymbol{C}^*$ such that the sorted eigenvalues are $\lambda_1 \geq \lambda_2 > 0 > \lambda_3$. This convention may be enforced by ensuring that $\det[\boldsymbol{C}^*] < 0$. The vector $\boldsymbol{v}_i$ is the unit eigenvector corresponding to the eigenvalue λ_i. Now, substituting the eigendecomposition of $\alpha \boldsymbol{C}^*$ into Eq. (171) yields

$$\alpha \boldsymbol{V} \boldsymbol{\Lambda} \boldsymbol{V}^T = \boldsymbol{A}_C^{-1} - \boldsymbol{r}_C \boldsymbol{r}_C^T \tag{174}$$

which may be rewritten as

$$\boldsymbol{V}^T \boldsymbol{A}_C^{-1} \boldsymbol{V} = \alpha \boldsymbol{\Lambda} + \boldsymbol{r}_\lambda \boldsymbol{r}_\lambda^T \tag{175}$$

where

$$\boldsymbol{r}_\lambda = \boldsymbol{V}^T \boldsymbol{r}_C = \begin{bmatrix} r_{\lambda_1} \\ r_{\lambda_2} \\ r_{\lambda_3} \end{bmatrix} \tag{176}$$

We observe that the eigenvalues of the left-hand side of Eq. (175) are unchanged by the orthonormal matrix $\boldsymbol{V}$,

$$\text{eig}[\boldsymbol{V}^T \boldsymbol{A}_C^{-1} \boldsymbol{V}] = \text{eig}[\boldsymbol{A}_C^{-1}] = \text{eig}[\boldsymbol{A}_P^{-1}] = \{d_i\}_{i=1}^3 \tag{177}$$

where the three eigenvalues of $\boldsymbol{A}_P^{-1}$ are sorted in descending order, $d_1 \geq d_2 \geq d_3 > 0$. Consequently,

$$\{d_i\}_{i=1}^3 = \text{eig}[\alpha \boldsymbol{\Lambda} + \boldsymbol{r}_\lambda \boldsymbol{r}_\lambda^T] \tag{178}$$

We observe the right-hand side of Eq. (178) to be the eigenvalues for a rank-one update of a diagonal matrix. Therefore, applying the well-known result from Golub [105], we find the bounds on the eigenvalues to be

$$\alpha\lambda_1 + r^2 \geq d_1 \geq \alpha\lambda_1 \geq d_2 \geq \alpha\lambda_2 \geq d_3 \geq \alpha\lambda_3 \tag{179}$$

where $r = \|\boldsymbol{r}_C\| = \|\boldsymbol{r}_\lambda\|$. Diagonalization of $\boldsymbol{C}^*$ provides the upper-bound in the chain of inequalities. We will return to this intermediate (and not yet fully simplified) result in a moment.

The lower-bound may be found by the diagonalization of $\boldsymbol{A}_P^{-1}$. To proceed, take the inverse of Eq. (28)

$$\boldsymbol{A}_C^{-1} = \boldsymbol{T}_C^P \boldsymbol{A}_P^{-1} \boldsymbol{T}_P^C \tag{180}$$

which, after substitution into Eq. (171) yields

$$\alpha \boldsymbol{C}^* = \boldsymbol{T}_C^P \left(\boldsymbol{A}_P^{-1} - \boldsymbol{r}_P \boldsymbol{r}_P^T \right) \boldsymbol{T}_P^C \tag{181}$$

It follows that that

$$\{\alpha\lambda_i\}_{i=1}^3 = \text{eig}\left[\alpha \boldsymbol{C}^*\right] = \text{eig}\left[\boldsymbol{A}_P^{-1} - \boldsymbol{r}_P \boldsymbol{r}_P^T\right] \tag{182}$$

Since $\boldsymbol{A}_P^{-1}$ is a diagonal matrix, we recognize the right-hand side of Eq. (182) to be another case of a rank-one update of a diagonal matrix. Thus, once again applying the results of Golub [105] to this problem, we find that

$$d_1 \geq \alpha\lambda_1 \geq d_2 \geq \alpha\lambda_2 \geq d_3 \geq \alpha\lambda_3 \geq d_3 - r^2 \tag{183}$$

which is identical to Eq. (179), but with the lower bound specified (instead of the upper bound).

Without loss of generality, define the principal axis frame of the celestial body with $d_1 = a^2$ along the x-direction, $d_2 = b^2$ along the y-direction, and $d_3 = c^2$ along the z-direction. Thus, as required, $\{d_i\}_{i=1}^3$ is a sorted list of a^2, b^2, c^2. It follows directly that $d_3 = c^2 > 0$. We also know that $d_3 - r^2 = c^2 - r^2 \leq 0$ since the camera must be outside the celestial body. Moreover, recall that our choice of

$\det[\boldsymbol{C}^*] < 0$ ensures that the eigenvalue of unique sign is λ_3 and that this eigenvalue is negative. It follows that $\alpha > 0$ and that $\alpha\lambda_3 < 0$. Therefore, combining Eq. (179) and Eq. (183), replacing d_1, d_2, d_3 with a^2, b^2, c^2, and incorporating the supplemental inequalities, the generic result becomes

$$\alpha\lambda_1 + r^2 \geq a^2 \geq \alpha\lambda_1 \geq b^2 \geq \alpha\lambda_2 \geq c^2 > 0 \geq \alpha\lambda_3 \geq c^2 - r^2 \tag{184}$$

Because a, b, c are known (since the body shape is known) and $\lambda_1, \lambda_2, \lambda_3$ are computed from the observed conic envelope $\boldsymbol{C}^*$, this chain of inequalities provides constraints on the possible values of the scale factor α. We remind the reader that α and r are the only unknowns in these inequalities.

B. FINDING SCALE FACTOR FOR DIFFERENT SHAPES

The key to solving the five degree-of-freedom pose problem is finding the unknown scale α. This is easily done for a sphere and spheroid using the inequalities from Eq. (184). The situation is more complicated for a triaxial ellipsoid.

1) SPHERE

For a sphere, $a = b = c > 0$ such that $d_i = a^2$ for $i = 1, 2, 3$. Therefore, the eigenvalue inequality becomes

$$\alpha\lambda_1 + r^2 \geq a^2 \geq \alpha\lambda_1 \geq a^2 \geq \alpha\lambda_2 \geq a^2 > 0 \geq \alpha\lambda_3 \geq a^2 - r^2 \tag{185}$$

From which we immediately conclude that

$$a^2 = \alpha\lambda_1 = \alpha\lambda_2 \neq \alpha\lambda_3 \tag{186}$$

and that

$$\alpha = a^2/\lambda_1 = a^2/\lambda_2 \tag{187}$$

In the presence of measurement noise, we often find that $\lambda_1 \neq \lambda_2$. When this is the case, we may compute α as the average value

$$\alpha = \frac{2a^2}{\lambda_1 + \lambda_2} \tag{188}$$

2) OBLATE SPHEROID

For an oblate spheroid, $a = b > c > 0$ such that $d_i = a^2$ for $i = 1, 2$. Therefore, the eigenvalue inequality becomes

$$\alpha\lambda_1 + r^2 \geq a^2 \geq \alpha\lambda_1 \geq a^2 \geq \alpha\lambda_2 \geq c^2 > 0 \geq \alpha\lambda_3 \geq c^2 - r^2 \tag{189}$$

Using the same approach as for a sphere, we conclude that

$$a^2 = \alpha\lambda_1 \tag{190}$$

which may be solved for α

$$\alpha = a^2/\lambda_1 \tag{191}$$

3) TRIAXIAL ELLIPSOID

For an oblate spheroid, $a > b > c > 0$ and the eigenvalue inequality becomes

$$\alpha\lambda_1 + r^2 \geq a^2 \geq \alpha\lambda_1 \geq b^2 \geq \alpha\lambda_2 \geq c^2 > 0 \geq \alpha\lambda_3 \geq c^2 - r^2 \tag{192}$$

No simplifications are possible. At best, one may determine a range of possible values for $\alpha \in [\alpha_{min}, \alpha_{max}]$, where

$$\alpha_{min} = \max\left[\frac{b^2}{\lambda_1}, \frac{c^2}{\lambda_2}\right] \tag{193}$$

$$\alpha_{max} = \min\left[\frac{a^2}{\lambda_1}, \frac{b^2}{\lambda_2}\right] \tag{194}$$

As we will see, each selection of α will lead to a different pose. This produces a one-dimensional manifold of pose values that may be parameterized by the scalar α.

C. COMPUTING POSE FROM A HORIZON PROJECTION

Assuming the scale factor α is known (unique for a sphere and spheroid, a range of values for an ellipsoid), we wish to compute the pose of the celestial body. To do this, we first estimate $\boldsymbol{r}_C$ and then compute the relative attitude using the appropriate attitude determination algorithm from Section IV-B.

To obtain a solution for $\boldsymbol{r}_C$, return to the diagonalized conic envelope equation from Eq. (175):

$$\boldsymbol{V}^T\boldsymbol{A}_C^{-1}\boldsymbol{V} = \alpha\boldsymbol{\Lambda} + \boldsymbol{r}_\lambda\boldsymbol{r}_\lambda^T \tag{195}$$

where $\boldsymbol{V}$ and $\boldsymbol{\Lambda}$ are from Eq. (173) and where $\boldsymbol{r}_\lambda$ is from Eq. (176). As discussed by Golub [105], this rank-one update of a diagonal matrix obeys the constraint

$$1 + \sum_{i=1}^{n} \frac{r_{\lambda_i}^2}{\alpha\lambda_i - d_j} = 0 \tag{196}$$

Expanding the summation, we have

$$\frac{r_{\lambda_1}^2}{d_j - \alpha\lambda_1} + \frac{r_{\lambda_2}^2}{d_j - \alpha\lambda_2} + \frac{r_{\lambda_3}^2}{d_j - \alpha\lambda_3} = 1 \tag{197}$$

where α, $\{\lambda_i\}_{i=1}^3$, and $\{d_j\}_{j=1}^3$ are all known. This may be used to solve for $\boldsymbol{r}_\lambda$. Since $\boldsymbol{V}$ is known and invertable, a solution for $\boldsymbol{r}_\lambda$ leads directly to a solution for $\boldsymbol{r}_C$. The procedure is slightly different for an oblate spheroid and for a triaxial ellipsoid, and the details are now presented. The sphere is a special case and is presented last.

1) OBLATE SPHEROID

For an oblate spheroid, we have $a = b > c$, which leads to the following two equations:

$$\frac{r_{\lambda_1}^2}{a^2 - \alpha\lambda_1} + \frac{r_{\lambda_2}^2}{a^2 - \alpha\lambda_2} + \frac{r_{\lambda_3}^2}{a^2 - \alpha\lambda_3} = 1 \tag{198}$$

$$\frac{r_{\lambda_1}^2}{c^2 - \alpha\lambda_1} + \frac{r_{\lambda_2}^2}{c^2 - \alpha\lambda_2} + \frac{r_{\lambda_3}^2}{c^2 - \alpha\lambda_3} = 1 \tag{199}$$

The first term in Eq. (198) would generally be problematic since $a^2 = \alpha\lambda_1$ [see Eq. (190)] for an oblate spheroid.

Fortunately, we observe that $r_{\lambda_1} = 0$ in this situation and the first term vanishes. Therefore, we only have

$$\frac{r_{\lambda_2}^2}{a^2-\alpha\lambda_2} + \frac{r_{\lambda_3}^2}{a^2-\alpha\lambda_3} = 1 \tag{200}$$

$$\frac{r_{\lambda_2}^2}{c^2-\alpha\lambda_2} + \frac{r_{\lambda_3}^2}{c^2-\alpha\lambda_3} = 1 \tag{201}$$

which may be used to construct a linear system

$$\begin{bmatrix} \frac{1}{a^2-\alpha\lambda_2} & \frac{1}{a^2-\alpha\lambda_3} \\ \frac{1}{c^2-\alpha\lambda_2} & \frac{1}{c^2-\alpha\lambda_3} \end{bmatrix} \boldsymbol{\rho} = 1_{2\times 1} \tag{202}$$

where

$$\boldsymbol{\rho} = \begin{bmatrix} \rho_1 \\ \rho_2 \end{bmatrix} = \begin{bmatrix} r_{\lambda_2}^2 \\ r_{\lambda_3}^2 \end{bmatrix} \tag{203}$$

Since this is a simple 2×2 system, it may be solved analytically for $\boldsymbol{\rho}$

$$\boldsymbol{\rho} = \frac{1}{\alpha(\lambda_2-\lambda_3)} \begin{bmatrix} -(c^2-\alpha\lambda_2)(a^2-\alpha\lambda_2) \\ (c^2-\alpha\lambda_3)(a^2-\alpha\lambda_3) \end{bmatrix} \tag{204}$$

After substitution for α from Eq. (191), this may be further simplified to

$$\boldsymbol{\rho} = \frac{a^2\lambda_1}{\lambda_2-\lambda_3} \begin{bmatrix} -((c/a)^2-\lambda_2/\lambda_1)(1-\lambda_2/\lambda_1) \\ ((c/a)^2-\lambda_3/\lambda_1)(1-\lambda_3/\lambda_1) \end{bmatrix} \tag{205}$$

Therefore, we may solve for $\boldsymbol{r}_C$ as

$$\boldsymbol{r}_C = \boldsymbol{V}\boldsymbol{r}_\lambda = \boldsymbol{V} \begin{bmatrix} 0 \\ \pm\sqrt{\rho_1} \\ \pm\sqrt{\rho_2} \end{bmatrix} \tag{206}$$

There are four possible choices for the signs of the terms in $\boldsymbol{r}_\lambda$. Two of these will produce relative positions with the observed celestial body in front of the camera. Either of these solutions perfectly satisfies the observed data and result in a twofold ambiguity.

A simple way to enforce this cheirality test and to find the two correct solutions is to define $\boldsymbol{r}'_{C_1}$ and $\boldsymbol{r}'_{C_2}$ as

$$\boldsymbol{r}'_{C_1} = \boldsymbol{V} \begin{bmatrix} 0 \\ \sqrt{\rho_1} \\ \sqrt{\rho_2} \end{bmatrix} \quad \text{and} \quad \boldsymbol{r}'_{C_2} = \boldsymbol{V} \begin{bmatrix} 0 \\ -\sqrt{\rho_1} \\ \sqrt{\rho_2} \end{bmatrix} \tag{207}$$

and then compute the two solutions as

$$\{\boldsymbol{r}_C\}_i = \text{sign}[\boldsymbol{k}^T\boldsymbol{r}'_{C_i}]\boldsymbol{r}'_{C_i}, \quad i = 1, 2 \tag{208}$$

where $\boldsymbol{k}^T = [0, 0, 1]$.

Once we have the two possible solutions for $\boldsymbol{r}_C$, we may find the spheroid's axis of revolution in the camera frame $\boldsymbol{w}_C$ by following the procedure in Sec. IV-B (and as codified in Algorithm 3).

This provides the fully general solution to the five degree-of-freedom pose problem with an oblate spheroid. It completely avoids creation of the arbitrary geometric constructs used in [104]. It also avoids the cubic solution required in Modenini's derivation, as well as the complicated trigonometric solutions shown in [104]. Here, we have reduced the solution to computing the eigenvalues/eigenvectors of $\boldsymbol{C}^*$, which permit a direct analytic solution for $\boldsymbol{r}_C$ and $\boldsymbol{w}_C$.

2) TRIAXIAL ELLIPSOID

For a triaxial ellipsoid, we have $a > b > c$, which leads to the following three equations:

$$\frac{r_{\lambda_1}^2}{a^2-\alpha\lambda_1} + \frac{r_{\lambda_2}^2}{a^2-\alpha\lambda_2} + \frac{r_{\lambda_3}^2}{a^2-\alpha\lambda_3} = 1 \tag{209}$$

$$\frac{r_{\lambda_1}^2}{b^2-\alpha\lambda_1} + \frac{r_{\lambda_2}^2}{b^2-\alpha\lambda_2} + \frac{r_{\lambda_3}^2}{b^2-\alpha\lambda_3} = 1 \tag{210}$$

$$\frac{r_{\lambda_1}^2}{c^2-\alpha\lambda_1} + \frac{r_{\lambda_2}^2}{c^2-\alpha\lambda_2} + \frac{r_{\lambda_3}^2}{c^2-\alpha\lambda_3} = 1 \tag{211}$$

The scalar α is chosen somewhere between α_{min} and α_{max}, with each choice for α producing a solution at a different range. This produces a one-dimensional manifold of possible pose solutions in the six degree-of-freedom state space.

Using the same procedure as for the spheroid, we stack the equations above to form a linear system,

$$\begin{bmatrix} \frac{1}{a^2-\alpha\lambda_1} & \frac{1}{a^2-\alpha\lambda_2} & \frac{1}{a^2-\alpha\lambda_3} \\ \frac{1}{b^2-\alpha\lambda_1} & \frac{1}{b^2-\alpha\lambda_2} & \frac{1}{b^2-\alpha\lambda_3} \\ \frac{1}{c^2-\alpha\lambda_1} & \frac{1}{c^2-\alpha\lambda_2} & \frac{1}{c^2-\alpha\lambda_3} \end{bmatrix} \boldsymbol{\rho} = 1_{3\times 1} \tag{212}$$

where

$$\boldsymbol{\rho} = \begin{bmatrix} \rho_1 \\ \rho_2 \\ \rho_3 \end{bmatrix} = \begin{bmatrix} r_{\lambda_1}^2 \\ r_{\lambda_2}^2 \\ r_{\lambda_3}^2 \end{bmatrix} \tag{213}$$

Since the 3×3 matrix in Eq. (212) is full rank, we may solve for $\boldsymbol{\rho}$ directly. Therefore, with $\boldsymbol{\rho}$ known

$$\boldsymbol{r}_C = \boldsymbol{V}\boldsymbol{r}_\lambda = \boldsymbol{V} \begin{bmatrix} \pm\sqrt{\rho_1} \\ \pm\sqrt{\rho_2} \\ \pm\sqrt{\rho_3} \end{bmatrix} \tag{214}$$

only four of which will produce an estimate with the celestial body in front of the camera (i.e., positive z-component of $\boldsymbol{r}_C$). We may find the four solutions using a similar procedure as for the oblate spheroid. First compute the four intermediate results

$$\boldsymbol{r}'_{C_1} = \boldsymbol{V} \begin{bmatrix} \sqrt{\rho_1} \\ \sqrt{\rho_2} \\ \sqrt{\rho_3} \end{bmatrix}, \quad \boldsymbol{r}'_{C_2} = \boldsymbol{V} \begin{bmatrix} -\sqrt{\rho_1} \\ \sqrt{\rho_2} \\ \sqrt{\rho_3} \end{bmatrix} \tag{215a}$$

$$\boldsymbol{r}'_{C_3} = \boldsymbol{V} \begin{bmatrix} \sqrt{\rho_1} \\ -\sqrt{\rho_2} \\ \sqrt{\rho_3} \end{bmatrix}, \quad \boldsymbol{r}'_{C_4} = \boldsymbol{V} \begin{bmatrix} -\sqrt{\rho_1} \\ -\sqrt{\rho_2} \\ \sqrt{\rho_3} \end{bmatrix} \tag{215b}$$

which permit the four solutions to be found by

$$\{\boldsymbol{r}_C\}_i = \text{sign}[\boldsymbol{k}^T\boldsymbol{r}'_{C_i}]\boldsymbol{r}'_{C_i}, \quad i = 1, 2, 3, 4 \tag{216}$$

where $\boldsymbol{k}^T = [0, 0, 1]$.

With these four possible values of $\boldsymbol{r}_C$ known, we may compute the relative attitude $\boldsymbol{T}_C^P$ following the procedure in Sec. IV-B (and as codified in Algorithm 4). For a triaxial ellipsoid, there are four possible attitudes for each position $\boldsymbol{r}_C$—leading to a total of $4 \times 4 = 16$ possible solutions. Alternatively, these 16 solutions correspond to eight different values of $\boldsymbol{r}_P$, with two possible attitudes at each unique location. All 16 of these solutions (either $4 \times 4 = 16$ in the camera frame or $8 \times 2 = 16$ in the celestial body frame) perfectly satisfy the original problem. It is impossible to distinguish between these equivalent solutions without additional information. Moreover, each of these 16 solutions belongs to its own 1D manifold as α varies between α_{min} and α_{max}.

3) SPHERE

While a similar approach may be taken for a sphere, we observe that this is an unwise choice in practice. If the celestial body is spherical, we can solve for $\boldsymbol{r}_C$ with the Christian-Robinson algorithm—and this solution will be better than what is achieved by estimating $\boldsymbol{r}_C$ from the horizon fit $\boldsymbol{C}$. There is no need to discuss solutions for $\boldsymbol{T}_C^P$ or $\boldsymbol{r}_P$, since neither of these can be found for a spherical body.

D. RELATIONSHIP BETWEEN SCALE FACTOR AND RANGE

It is also possible to independently compute the range without computing the full relative position. It is important to stress that we never have to do this in order to compute pose. The present discussion is included purely for geometric insight and academic interest.

Beginning with the assumption that the scale factor α is known, it is possible to solve directly for the unknown range. This may be achieved by taking the trace of both sides of Eq. (171)

$$\alpha \mathrm{Tr}\left[\boldsymbol{C}^*\right] = \mathrm{Tr}\left[\boldsymbol{A}_C^{-1}\right] - \mathrm{Tr}\left[\boldsymbol{r}_C \boldsymbol{r}_C^T\right] \tag{217}$$

which becomes

$$\alpha\left(\lambda_1 + \lambda_2 + \lambda_3\right) = a^2 + b^2 + c^2 - r^2 \tag{218}$$

This is exactly the same as Eq. (131) after some trivial rearrangement. Directly solving for r^2, we obtain

$$r^2 = a^2 + b^2 + c^2 - \alpha\left(\lambda_1 + \lambda_2 + \lambda_3\right) \tag{219}$$

Simple analytic expressions may be found for the range in the case of a sphere and spheroid. These are now derived.

1) SPHERE

For a sphere, we know that

$$a^2 = b^2 = c^2 = \alpha\lambda_1 = \alpha\lambda_2$$

Thus, substituting these relations into Eq. (219),

$$r^2 = a^2 + a^2 + a^2 - \left(a^2 + a^2 + \alpha\lambda_3\right) = a^2 - \alpha\lambda_3 \tag{220}$$

which is easy to solve for r,

$$r = \sqrt{a^2 - \alpha\lambda_3} \tag{221}$$

Since $\alpha = a^2/\lambda_1$ for a sphere, this becomes

$$r = a\sqrt{1 - \lambda_3/\lambda_1} \tag{222}$$

2) OBLATE SPHEROID

For an oblate spheroid, we know that

$$a^2 = b^2 = \alpha\lambda_1 \tag{223}$$

Thus, substituting these relations into Eq. (219),

$$\begin{aligned} r^2 &= a^2 + a^2 + c^2 - \left(a^2 + \alpha\lambda_2 + \alpha\lambda_3\right) \\ &= a^2 + c^2 - \alpha\lambda_2 - \alpha\lambda_3 \end{aligned} \tag{224}$$

Now, substituting $\alpha = a^2/\lambda_1$ from Eq. (191)

$$r^2 = a^2 + c^2 - a^2\frac{\lambda_2}{\lambda_1} - a^2\frac{\lambda_3}{\lambda_1} \tag{225}$$

$$r^2 = a^2\left(1 + \frac{c^2}{a^2} - \frac{\lambda_2}{\lambda_1} - \frac{\lambda_3}{\lambda_1}\right) \tag{226}$$

While working with the ratio c/a is easy enough, reference spheroids for celestial bodies are often defined in terms of their flattening, f, which is defined as [42]

$$f = \frac{a - c}{a} = 1 - \frac{c}{a} \tag{227}$$

Thus, it is trivial to rewrite our expression for r directly in terms of the flattening parameter that is more commonly reported for shape models

$$r^2 = a^2\left(2 + f^2 - 2f - \frac{\lambda_2}{\lambda_1} - \frac{\lambda_3}{\lambda_1}\right) \tag{228}$$

Since it is guaranteed that $r > 0$, we have either

$$r = a\sqrt{1 + \frac{c^2}{a^2} - \frac{\lambda_2}{\lambda_1} - \frac{\lambda_3}{\lambda_1}} \tag{229}$$

or

$$r = a\sqrt{2 + f^2 - 2f - \frac{\lambda_2}{\lambda_1} - \frac{\lambda_3}{\lambda_1}} \tag{230}$$

As a final aside, we note that (in contrast to [104]) there is no need to explicitly transform into a local-horizontal/local-vertical (LVLH) frame or to explicitly keep track of latitude in these derivations.

3) TRIAXIAL ELLIPSOID

In the case of a triaxial ellipsoid, the original result from Eq. (219) does not simplify in general. Therefore, the range is computed as

$$r = \sqrt{a^2 + b^2 + c^2 - \alpha\left(\lambda_1 + \lambda_2 + \lambda_3\right)} \tag{231}$$

We note the explicit dependence on the scale factor α.

E. ALGORITHM SUMMARY

The results above may be assembled into straightforward algorithms for horizon-based pose estimation. The solution for an oblate spheroid is given in Algorithm 5, while the solution for a triaxial ellipsoid is given in Algorithm 6. The efficacy of these two algorithms are best demonstrated by an example.

For the case of an oblate spheroid, once again consider the example of Ceres as imaged by a spacecraft at a range of 10,000 km. We assume the same visible spectrum camera as in other examples. Under the conditions in this example, Ceres projects to an image ellipse with semimajor axis of 285.58 pixels and semiminor axis of 282.75 pixels. Results from a 10,000-run Monte Carlo simulation are shown in Tables 9–12. Table 9 and Table 10 show performance statistics for the position component of the pose solution. Table 11 and Table 12 show performance statistics for the attitude component of the pose solution.

As expected, position estimation is worst along the camera boresight direction. There does not appear to be a strong preference in the direction of largest error for the orientation of the celestial body's axis of symmetry. These trends in covariance shape are similar to the position-only solution from Sec. III and the attitude-only solution for a spheroid with known $\boldsymbol{r}_C$ from Sec. IV-B.

Algorithm 5 Pseudocode for pose algorithm with horizon fit of an oblate spheroid

procedure $[\{\boldsymbol{r}_C\}, \{\boldsymbol{w}_C\}] = \text{Pose\_ObSp}(\boldsymbol{C}^*, a, c)$
  if $\det[\boldsymbol{C}^*] > 0$ **then**
    $\boldsymbol{C}^* = -\boldsymbol{C}^*$
  compute $[\boldsymbol{\Lambda}, \boldsymbol{V}] = \text{eig}[\boldsymbol{C}^*]$ ▷ Eq. (173)
  compute $\boldsymbol{\rho}$ ▷ Eq. (205)
  for $i = 1$ to 2 **do**
    compute $\boldsymbol{r}'_{C_i}$ ▷ Eq. (207)
    compute $\{\boldsymbol{r}_C\}_i$ ▷ Eq. (208)
    compute $\{\boldsymbol{w}_C\}_i$ ▷ Algorithm 3
return $\{\boldsymbol{r}_C\}_{i=1}^2, \{\boldsymbol{w}_C\}_{i=1}^2$

Algorithm 6 Pseudocode for pose algorithm with horizon fit of a triaxial ellipsoid

procedure $[\{\boldsymbol{r}_C\}, \{\boldsymbol{T}_C^P\}] = \text{Pose\_El}(\boldsymbol{C}^*, a, b, c)$
  if $\det[\boldsymbol{C}^*] > 0$ **then**
    $\boldsymbol{C}^* = -\boldsymbol{C}^*$
  compute $[\boldsymbol{\Lambda}, \boldsymbol{V}] = \text{eig}[\boldsymbol{C}^*]$ ▷ Eq. (173)
  compute α_{min} and $\alpha_{\max}$ ▷ Eqs. (193) & (194)
  choose $\alpha \in [\alpha_{min}, \alpha_{max}]$
  compute $\boldsymbol{\rho}$ ▷ Eq. (212)
  for $i = 1$ to 4 **do**
    compute $\boldsymbol{r}'_{C_i}$ ▷ Eq. (215)
    compute $\{\boldsymbol{r}_C\}_i$ ▷ Eq. (216)
    compute $\{\{\boldsymbol{T}_C^P\}_{j=1}^4\}_i$ ▷ Algorithm 4
return $\{\boldsymbol{r}_C\}_{i=1}^4, \{\{\boldsymbol{T}_C^P\}_{j=1}^4\}_{i=1}^4$

TABLE 9. Pose estimation performance statistics (position component) for synthetic images of the dwarf planet Ceres. Pose estimation performed using Algorithm 5. Statistics are computed from a 10,000-run Monte Carlo simulation, with each algorithm operating on the same set of simulation inputs.

Method	Mean Error (km)	Standard Deviation (km)
Non-Iterative Ellipse Methods		
Modified Fitzgibbon [62], [63]	0.3305	2.6647
Semihyper Least-Squares [60]	0.0910	2.6976
Hyper Least-Squares [60]	0.0910	2.6976
Iterative Ellipse Methods		
Doubly-Optimal Fit [66]	0.0911	2.6976
Fund. Numerical Scheme [65]	0.0901	2.6974
Szpak, et al. [67]	0.0117	2.6895

TABLE 10. Pose estimation covariance (position component) for synthetic images of the dwarf planet Ceres. The root-sum-square (RSS) of these values are equivalent to the right-most column in Table 9. Errors are expressed in the camera frame.

Method	σ_x (km)	σ_y (km)	σ_z (km)
Non-Iterative Ellipse Methods			
Modified Fitzgibbon [62], [63]	0.2039	0.0701	2.6560
Semihyper Least-Squares [60]	0.2079	0.0714	2.6886
Hyper Least-Squares [60]	0.2079	0.0714	2.6886
Iterative Ellipse Methods			
Doubly-Optimal Fit [66]	0.2079	0.0714	2.6886
Fund. Numerical Scheme [65]	0.2078	0.0714	2.6884
Szpak, et al. [67]	0.2069	0.0711	2.6806

TABLE 11. Pose estimation performance statistics (attitude component) for synthetic images of the dwarf planet Ceres. Results indicate errors in the direction of the oblate spheroid axis of symmetry, $\boldsymbol{w}_C$. Pose estimation performed using Algorithm 5. Statistics are computed from a 10,000-run Monte Carlo simulation, with each algorithm operating on the same set of simulation inputs.

Method	Mean Error (deg)	Standard Deviation (deg)
Non-Iterative Ellipse Methods		
Modified Fitzgibbon [62], [63]	0.1173	0.7401
Semihyper Least-Squares [60]	0.0248	0.7402
Hyper Least-Squares [60]	0.0248	0.7402
Iterative Ellipse Methods		
Doubly-Optimal Fit [66]	0.0248	0.7402
Fund. Numerical Scheme [65]	0.0248	0.7402
Szpak, et al. [67]	0.0396	0.7401

TABLE 12. Pose estimation covariance (attitude component) for synthetic images of the dwarf planet Ceres. The root-sum-square (RSS) of these values are equivalent to the right-most column in Table 11. Errors are expressed in the camera frame.

Method	σ_x (deg)	σ_y (deg)	σ_z (deg)
Non-Iterative Ellipse Methods			
Modified Fitzgibbon [62], [63]	0.6633	0.3209	0.0686
Semihyper Least-Squares [60]	0.6606	0.3265	0.0704
Hyper Least-Squares [60]	0.6606	0.3265	0.0704
Iterative Ellipse Methods			
Doubly-Optimal Fit [66]	0.6606	0.3265	0.0704
Fund. Numerical Scheme [65]	0.6605	0.3265	0.0704
Szpak, et al. [67]	0.6612	0.3251	0.0699

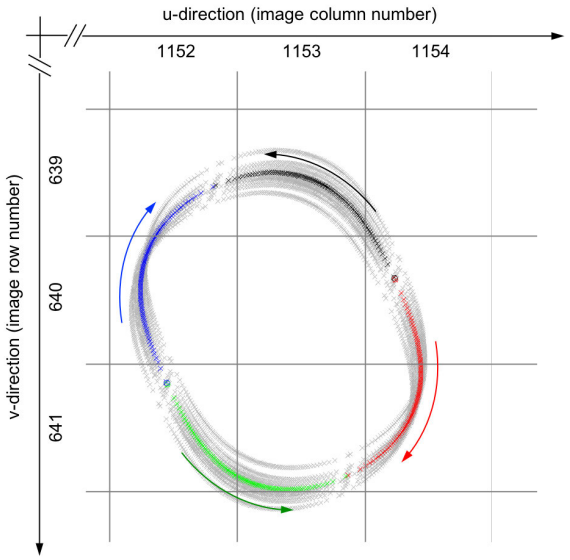

FIGURE 19. Visualization of 1D manifold of solutions for a triaxial ellipsoid using synthetic images of Miamis. The red, green, blue, and black markers show the projection of the four noise-free solutions for r_C onto the image, with arrows indicating the progression along the manifold as α varies from α_{min} to α_{max}. The gray markers show the results from 10 Monte Carlo runs with a limb localization error of $\sigma = 0.07$ pixel. Gridlines show pixel boundaries. Results produced using Algorithm 6.

For the case of a triaxial ellipsoid, consider synthetic images of Miamis from a range of 4,000 km using the same visible spectrum camera as in other examples. Under the conditions in this particular example, Miamis projects to an image ellipse with semimajor axis of 291.39 pixels and semiminor axis of 286.91 pixels. It is difficult to meaningfully display perturbations of the 16 possible 1D manifolds in six degree-of-freedom state space. Instead, consider only the projection of $\boldsymbol{r}_C$ onto the image plane. We can track the evolution of the bearing to the body center in pixel space as the scale factor α goes from α_{min} to α_{max}. These results are shown in Fig. 19. The range from the camera to the center of Miamis varies from 3,940.6 km to 4,000.2 km over this range of α. We can see that it is possible for the apparent direction to the body to change by a few pixels, depending on (1) the choice of α and (2) which of the four solutions of $\boldsymbol{r}_C$ is selected. The magnitude of this effect will be different for different viewing geometries of Miamis or for observations of other celestial bodies. In some situations these variations may be small enough to neglect, but (as this example highlights) this is not always the case.

VI. CONCLUSION

The apparent horizon of a celestial body in digital image may be used for spacecraft navigation. Since most large celestial bodies are well-modeled at the global level by a sphere, spheroid, or triaxial ellipsoid, the horizon projects to a conic in the image. Depending on the specific body and viewing geometry, the horizon arc may appear as a circle, ellipse, parabola, or hyperbola. This work shows how the projected conic (usually an ellipse) may be used to estimate the relative position, relative attitude, or both. Algorithms are presented for all three of these state estimation scenarios and for all three body shapes (sphere, spheroid, triaxial ellipsoid). This work is the first time that all of these closely related problems have been studied in a holistic manner and within a common mathematical framework.

ACKNOWLEDGMENT

The author thanks Chris D'Souza, Randy Christensen, and Mike Kudenov for valuable discussions and feedback that improved the quality of the final manuscript. He also thanks Ali Al-Sharadqah for helpful discussions on the topic of ellipse fitting and for sharing prototype code for the algorithms in Ref. [66].

REFERENCES

[1] J. A. Christian, J. McMahon, D. N. DellaGuistina, C. M. Ernst, R. P. Russell, D. R. Golish, J. S. McCabe, S. Keller, D. S. Lauretta, P. Gay, and P. Schoch, "Resolved imagery as a tool for space science and exploration," in *Proc. NASA Explor. Sci. Forum*, NASA Ames Research Center, 2019.

[2] W. Owen, T. Duxbury, C. Acton, S. Synnott, J. Riedel, and S. Bhaskaran, "A brief history of optical navigation at JPL," in *Proc. AAS Guid. Control Conf.*, 2008, AAS Paper 08-053.

[3] W. M. Owen, "Methods of optical navigation," in *Proc. AAS/AIAA Space Flight Mech. Meeting*, 2011, AAS Paper 11-215.

[4] D. Mortari, C. N. D'Souza, and R. Zanetti, "Image processing of illuminated ellipsoid," *J. Spacecraft Rockets*, vol. 53, no. 3, pp. 448–456, May 2016, doi: 10.2514/1.A33342.

[5] J. A. Christian, "Accurate planetary limb localization for image-based spacecraft navigation," *J. Spacecraft Rockets*, vol. 54, no. 3, pp. 708–730, May 2017, doi: 10.2514/1.A33692.

[6] C. Hanak, T. Crain, and R. Bishop, "Crater identification algorithm for the lost in low lunar orbit scenario," in *Proc. AAS Guid. Control Conf.*, 2008, AAS Paper 10-052.

[7] B. Maass, S. Woicke, W. M. Oliveira, B. Razgus, and H. Krüger, "Crater navigation system for autonomous precision landing on the moon," *J. Guid., Control, Dyn.*, vol. 43, no. 8, pp. 1414–1431, Aug. 2020, doi: 10.2514/1.G004850.

[8] J. A. Christian, H. Derksen, and R. Watkins, "Lunar crater identification in digital images," submitted for publication, *J. Astron. Sci.*, pp. 1–94, Sep. 2020. [Online]. Available: https://arxiv.org/abs/2009.01228

[9] R. W. Gaskell, O. S. Barnouin-Jha, D. J. Scheeres, A. S. Konopliv, T. Mukai, S. Abe, J. Saito, M. Ishiguro, T. Kubota, T. Hashimoto, J. Kawaguchi, M. Yoshikawa, K. Shirakawa, T. Kominato, N. Hirata, and H. Demura, "Characterizing and navigating small bodies with imaging data," *Meteoritics Planet. Sci.*, vol. 43, no. 6, pp. 1049–1061, Jun. 2008, doi: 10.1111/j.1945-5100.2008.tb00692.x.

[10] D. Adams, T. B. Criss, and U. J. Shankar, "Passive optical terrain relative navigation using APLNav," in *Proc. IEEE Aerosp. Conf.*, Mar. 2008, pp. 1–9, doi: 10.1109/AERO.2008.4526303.

[11] R. Olds, A. May, C. Mario, R. Hamilton, C. Debrunner, and K. Anderson, "The application of optical based feature tracking to OSIRIS-REx asteroid sample collection," in *Proc. AAS Guid., Navigat., Control Conf.*, 2015, AAS Paper 15-124.

[12] A. Johnson, N. Villaume, C. Umsted, A. Kourchians, D. Sternberg, N. Trawny, Y. Cheng, E. Geipel, and J. Montgomery, "The Mars 2020 lander vision system field test," in *Proc. AAS Guid., Navigat. Control Conf.*, 2020, AAS Paper 20-105.

[13] J. A. Christian, L. Hong, P. McKee, R. Christensen, and T. P. Crain, "Image-based lunar terrain relative navigation without a map: Measurements," *J. Spacecraft Rockets*, vol. 58, no. 1, pp. 164–181, Jan. 2021, doi: 10.2514/1.A34875.

[14] R. W. Astheimer, "High precision attitude determination by sensing the earth and lunar horizon in the infrared," *Automatica*, vol. 7, no. 1, pp. 83–88, Jan. 1971, doi: 10.1016/0005-1098(71)90082-3.

[15] J. R. Wertz, *Spacecraft Attitude Determination and Control*. Dordrecht, The Netherlands: D. Reidel, 1978.

[16] T. Nguyen, K. Cahoy, and A. Marinan, "Attitude determination for small satellites with infrared earth horizon sensors," *J. Spacecraft Rockets*, vol. 55, no. 6, pp. 1466–1475, Nov. 2018, doi: 10.2514/1.A34010.
[17] D. Modenini, A. Locarini, and M. Zannoni, "Attitude sensor from ellipsoid observations: A numerical and experimental validation," *Sensors*, vol. 20, no. 2:433, Jan. 2020, doi: 10.3390/s20020433.
[18] R. A. Harbison, P. C. Thomas, and P. C. Nicholson, "Rotational modeling of hyperion," *Celestial Mech. Dyn. Astron.*, vol. 110, no. 1, pp. 1–16, May 2011, doi: 10.1007/s10569-011-9337-3.
[19] R. Gupta and R. I. Hartley, "Linear pushbroom cameras," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 9, pp. 963–975, Sep. 1997, doi: 10.1109/34.615446.
[20] C. Leinert and D. Klüppelberg, "Stray light suppression in optical space experiments," *Appl. Opt.*, vol. 13, no. 3, pp. 556–564, 1974, doi: 10.1364/AO.13.000556.
[21] J. Arnoux, "Star sensor baffle optimization: Some helpful practical design rules," *Proc. SPIE*, vol. 2864, pp. 333–338, Nov. 1996, doi: 10.1117/12.258324.
[22] D. J. Schroeder, *Astronomical Optics*, 2nd ed. San Diego, CA, USA: Academic, 2000.
[23] J. A. Christian, L. Benhacine, J. Hikes, and C. D'Souza, "Geometric calibration of the orion optical navigation camera using star field images," *J. Astron. Sci.*, vol. 63, no. 4, pp. 335–353, Dec. 2016, doi: 10.1007/s40295-016-0091-3.
[24] B. J. Bos *et al.*, "In-flight calibration and performance of the OSIRIS-REx touch and go camera system (TAGCAMS)," *Space Sci. Rev.*, vol. 216, no. 71, pp. 1–52, Jun. 2020, doi: 10.1007/s11214-020-00682-x.
[25] S. Wu and H. Moore, "Experimental photogrammetry of lunar images," U.S. Government Printing Office, Washington, DC, USA, Geological Surv. Prof. Paper 1046-D, 1980, doi: 10.3133/pp1046D.
[26] G. Holst and T. Lomheim, *CMOS/CCD Sensors and Camera Systems*. Winter Park, FL, USA: SPIE Press, 2007.
[27] Y. Ma, S. Soatto, J. Košecká, and S. Sastry, *An Invitation to 3-D Vision: From Images to Geometric Models*. New York, NY, USA: Springer, 2010.
[28] J. A. Christian, "Optical navigation for a spacecraft in a planetary system," Ph.D. dissertation, Univ. Texas Austin, Austin, TX, USA, 2010.
[29] H. Sun, *Basic Optical Engineering for Engineers and Scientists*. Bellingham, WA, USA: SPIE Press, 2019.
[30] J. Gallier, *Geometric Methods and Applications*, 2nd ed. New York, NY, USA: Springer-Verlag, 2011.
[31] R. Zanetti, "Rotations, transformations, left quaternions, right quaternions?" *J. Astron. Sci.*, vol. 66, pp. 361–381, Sep. 2019, doi: 10.1007/s40295-018-00151-2.
[32] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 11, pp. 1330–1334, Nov. 2000, doi: 10.1109/34.888718.
[33] J. A. Christian and J. L. Crassidis, "Star identification and attitude determination with projective cameras," *IEEE Access*, 2021, doi: 10.1109/ACCESS.2021.3054836.
[34] C. Hash, "MESSENGER MDIS CALIBRATED (CDR) DATA E/V/H V1.0," NASA Planet. Data Syst., 2008.
[35] A. Nathues, H. Sierks, J. R. P. Gutierrez-Marques, 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 Planet. Data Syst., 2000.
[36] T. F. Thaller, "GALILEO ORBITAL OPERATIONS SOLID STATE IMAGING RAW EDR V1.0, GO-J/JSA-SSI-2-REDR-V1.0," NASA Planet. Data Syst., 2000.
[37] C. P. Porco, "CASSINI ORBITER SATURN ISSNA/ISSWA 2 EDR VERSION 1.0, CO-S-ISSNA/ISSWA-2-EDR-V1.0," NASA Planet. Data Syst., 2005.
[38] B. A. Smith *et al.*, "Voyager 2 at Neptune: Imaging science results," *Science*, vol. 246, no. 4936, pp. 1422–1449, 1989, doi: 10.1126/science.246.4936.1422.
[39] M. R. Showalter, M. K. Gordon, and D. Olson, "VG2 NEPTUNE ISS PROCESSED IMAGES V1.0, VGISS_8201-8210," NASA Planet. Data Syst., 2016.
[40] A. Cheng, "NEW HORIZONS Raw LORRI PLUTO ENCOUNTER V2.0, NH-P-LORRI-2-PLUTO-V2.0," NASA Planet. Data Syst., 2016.
[41] H. J. Melosh, *Planetary Surface Processes, Chapter the Shapes of Planets and Moons*. Cambridge, U.K.: Cambridge Univ. Press, 2011, pp. 25–48, doi: 10.1017/CBO9780511977848.003.
[42] W. Torge and J. Müller, *Geodesy*, 4th ed. Berlin, Germany: De Gruyter, 2012.
[43] I. Newton, *Philosophiæ Naturalis Principia Mathematica*. 1687.
[44] J. G. Semple and G. T. Kneebone, *Algebraic Projective Geometry*. Oxford, U.K.: Oxford Univ. Press, 1952.
[45] R. Hartley and A. Zisserman, *Multiple View Geometry*, 2nd ed. Cambridge, U.K.: Cambridge Univ. Press, 2003.
[46] J. A. Christian, "Optical navigation using planet's centroid and apparent diameter in image," *J. Guid., Control, Dyn.*, vol. 38, no. 2, pp. 192–204, Feb. 2015, doi: 10.2514/1.G000872.
[47] J. A. Christian, "Optical navigation using iterative horizon reprojection," *J. Guid., Control, Dyn.*, vol. 39, no. 5, pp. 1092–1103, May 2016, doi: 10.2514/1.G001569.
[48] C. Hollenberg, J. Christian, S. Bhaskaran, and W. Owen, "Centroiding performance for horizon-based optical navigation with Cassini images of Dione and Rhea," in *Proc. AAS/AIAA Spaceflight Mech. Meeting*, 2019, AAS Paper 19-494.
[49] D. Marr and E. Hildreth, "Theory of edge detection," *Proc. Roy. Soc. London B, Biol. Sci.*, vol. 207, no. 1167, pp. 187–217, Feb. 1980.
[50] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-8, no. 6, pp. 679–698, Nov. 1986, doi: 10.1109/TPAMI.1986.4767851.
[51] R. C. Gonzales and R. E. Woods, *Digital Image Processing*, 3rd ed. Upper Saddle River, NJ, USA: Prentice-Hall, 2008, pp. 706–728.
[52] A. Trujillo-Pino, K. Krissian, M. Alemán-Flores, and D. Santana-Cedrés, "Accurate subpixel edge location based on partial area effect," *Image Vis. Comput.*, vol. 31, no. 1, pp. 72–90, Jan. 2013, doi: 10.1016/j.imavis.2012.10.005.
[53] D. T. Renshaw and J. A. Christian, "Subpixel localization of isolated edges and streaks in digital images," *J. Imag.*, vol. 6, no. 5:33, May 2020, doi: 10.3390/jimaging6050033.
[54] D. Mortari, "Moon-sun attitude sensor," *J. Spacecraft Rockets*, vol. 34, no. 3, pp. 360–364, May 1997, doi: 10.2514/2.3217.
[55] J. Enright, "Moon-tracking modes for star trackers," *J. Guid., Control, Dyn.*, vol. 33, no. 1, pp. 171–185, Jan. 2010, doi: 10.2514/1.42716.
[56] J. E. Hansen and L. D. Travis, "Light scattering in planetary atmospheres," *Space Sci. Rev.*, vol. 16, no. 4, pp. 527–610, 1974, doi: 10.1007/BF00168069.
[57] E. J. McCartney, *Optics of the Atmosphere: Scattering by Molecules and Particles*. New York, NY, USA: Wiley, 1976, pp. 1–49.
[58] D. G. Hoag, "The history of Apollo onboard guidance, navigation, and control," *J. Guid., Control, Dyn.*, vol. 6, no. 1, pp. 4–13, Jan. 1983, doi: 10.2514/3.19795.
[59] J. A. Christian, "Horizon-based optical navigation using images of a planet with an atmosphere," in *Proc. AIAA/AAS Astrodynamics Specialist Conf.*, Sep. 2016, AIAA Paper 2016-5442, doi: 10.2514/6.2016-5442.
[60] K. Kanatani and P. Rangarajan, "Hyper least squares fitting of circles and ellipses," *Comput. Statist. Data Anal.*, vol. 55, no. 6, pp. 2197–2208, Jun. 2011, doi: 10.1016/j.csda.2010.12.012.
[61] G. Taubin, "Estimation of planar curves, surfaces, and nonplanar space curves defined by implicit equations with applications to edge and range image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 11, pp. 1115–1138, Nov. 1991, doi: 10.1109/34.103273.
[62] A. Fitzgibbon, M. Pilu, and R. B. Fisher, "Direct least square fitting of ellipses," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 5, pp. 476–480, May 1999, doi: 10.1109/34.765658.
[63] R. Halíř and J. Flusser, "Numerically stable direct least squares fitting of ellipses," in *Proc. 6th Int. Conf. Central Eur. Comput. Graph. Vis.*, 1998, pp. 125–132.
[64] A. Al-Sharadqah and N. Chernov, "Error analysis for circle fitting algorithms," *Electron. J. Statist.*, vol. 3, pp. 886–911, 2009, doi: 10.1214/09-EJS419.
[65] W. Chojnacki, M. J. Brooks, A. van den Hengel, and D. Gawley, "On the fitting of surfaces to data with covariances," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 11, pp. 1294–1303, Nov. 2000, doi: 10.1109/34.888714.
[66] A. Al-Sharadqah and N. Chernov, "A doubly optimal ellipse fit," *Comput. Statist. Data Anal.*, vol. 56, no. 9, pp. 2771–2781, Sep. 2012, doi: 10.1016/j.csda.2012.02.028.
[67] Z. L. Szpak, W. Chojnacki, and A. van der Hengel, "Guaranteed ellipse fitting with a confidence region and an uncertainty measure for centre, axes, and orientation," *J. Math. Imag.*, vol. 52, pp. 173–199, Sep. 2015, doi: 10.1007/s10851-014-0536-x.
[68] Z. L. Szpak, W. Chojnacki, and A. van den Hengel, "A comparison of ellipse fitting methods and implications for multiple-view geometry estimation," in *Proc. Int. Conf. Digit. Image Comput. Techn. Appl. (DICTA)*, Dec. 2012, pp. 1–8, doi: 10.1109/DICTA.2012.6411722.

[69] M. Feissel and F. Mignard, "The adoption of ICRS on 1 January 1998: Meaning and consequences," *Astron. Astrophys.*, vol. 331, pp. L33–L36, Mar. 1998.
[70] C. Ma, E. F. Arias, T. M. Eubanks, A. L. Fey, A.-M. Gontier, C. S. Jacobs, O. J. Sovers, B. A. Archinal, and P. Charlot, "The international celestial reference frame as realized by very long baseline interferometry," *Astronomical J.*, vol. 116, no. 1, pp. 516–546, Jul. 1998, doi: 10.1086/300408.
[71] A. L. Fey *et al.*, "The second international celestial reference frame by very long baseline interferometry," *Astronomical J.*, vol. 150, no. 58, pp. 1–16, 2015, doi: 10.1088/0004-6256/150/2/58.
[72] C. H. Acton, "Ancillary data services of NASA's navigation and ancillary information facility," *Planet. Space Sci.*, vol. 44, no. 1, pp. 65–70, Jan. 1996, doi: 10.1016/0032-0633(95)00107-7.
[73] C. Acton, N. Bachman, B. Semenov, and E. Wright, "A look towards the future in the handling of space science mission geometry," *Planet. Space Sci.*, vol. 150, pp. 9–12, Jan. 2018, doi: 10.1016/j.pss.2017.02.013.
[74] C. C. Liebe, "Star trackers for attitude determination," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 10, no. 6, pp. 10–16, Jun. 1995, doi: 10.1109/62.387971.
[75] C. C. Liebe, "Accuracy performance of star trackers—A tutorial," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 38, no. 2, pp. 587–599, Apr. 2002, doi: 10.1109/TAES.2002.1008988.
[76] B. A. Lampkin, "Sextant sighting performance for space navigation using simulated and real celestial targets," *Navigation*, vol. 12, no. 4, pp. 312–320, Dec. 1965, doi: 10.1002/j.2161-4296.1965.tb02149.x.
[77] T. R. Jorris and A. E. Barth, "The USAF manned space navigation experiment on Apollo and its implications on advanced manned spacecraft," in *Proc. Inst. Navigat. Nat. Space Meeting*, 1968, pp. 219–240.
[78] D. W. Smith and B. A. Lampkin, "Sextant sighting measurements from on board the Gemini XII spacecraft," Nat. Aeronaut. Space Admin., Washington, DC, USA, Tech. Rep. NASA TN D-4952, Dec. 1968.
[79] S. W. Powers, "Preliminary results from Skylab experiment T002, manual navigation," *Navigation*, vol. 21, no. 4, pp. 279–287, Dec. 1974, doi: 10.1002/j.2161-4296.1974.tb01227.x.
[80] R. Walsh and J. Ferguson Jr., "The United States Air Force manual space navigation experiment on Skylab/DOD/NASA Skylab experiment T-002," in *Proc. AIAA Astrodynamics Conf.*, Aug. 1976, p. 831, doi: 10.2514/6.1976-831.
[81] T. C. Duxbury and W. G. Breckenridge, "Mariner Mars 1969 optical approach navigation," in *Proc. 8th Aerosp. Sci. Meeting*, Jan. 1970, p. 70, doi: 10.2514/6.1970-70.
[82] N. Jerath and H. Ohtakay, "Mariner IX optical navigation using Mars lit limb," *J. Spacecraft Rockets*, vol. 11, no. 7, pp. 505–511, Jul. 1974, doi: 10.2514/3.62114.
[83] J. Campbell, S. Synnott, and G. Bierman, "Voyager orbit determination at Jupiter," *IEEE Trans. Autom. Control*, vol. 28, no. 3, pp. 256–268, Mar. 1983, doi: 10.1109/TAC.1983.1103223.
[84] S. Gillam, W. Owen, A. Vaughan, T.-C. Wang, J. Costello, R. Jacobson, D. Bluhm, J. Pojman, and R. Ionasescu, "Optical navigation for the Cassini/Huygens mission," in *Proc. AAS/AIAA Astrodynamics Specialist Conf.*, 2007, AAS Paper 07-252.
[85] W. M. Owen, P. J. Dumont, and C. D. Jackman, "Optical navigation preparations for New Horizons Pluto flyby," in *Proc. 23rd Int. Symp. Space Flight Dyn. (ISSFD)*, 2012, pp. 1–15.
[86] L. Lindegren, "Meridian observations of planets with a photoelectric multislit micrometer," *Astron. Astrophys.*, vol. 57, pp. 55–72, May 1977.
[87] M. Kaasalainen and P. Tanga, "Photocentre offset in ultraprecise astrometry: Implications for barycentre determination and asteroid modelling," *Astron. Astrophys.*, vol. 416, no. 1, pp. 367–373, Mar. 2004, doi: 10.1051/0004-6361:20031711.
[88] K. Muinonen and K. Lumme, "Disk-integrated brightness of a Lommel-Seeliger scattering ellipsoidal asteroid," *Astron. Astrophys.*, vol. 584, pp. 1–6, Dec. 2015, doi: 10.1051/0004-6361/201526456.
[89] J. A. Christian and S. B. Robinson, "Noniterative horizon-based optical navigation by Cholesky factorization," *J. Guid., Control, Dyn.*, vol. 39, no. 12, pp. 2757–2765, Dec. 2016, doi: 10.2514/1.G000539.
[90] I. Markovsky and S. Van Huffel, "Overview of total least-squares methods," *Signal Process.*, vol. 87, no. 10, pp. 2283–2302, Oct. 2007, doi: 10.1016/j.sigpro.2007.04.004.
[91] J. L. Crassidis and Y. Cheng, "Maximum likelihood analysis of the total least squares problem with correlated errors," *J. Guid., Control, Dyn.*, vol. 42, no. 6, pp. 1204–1217, Jun. 2019, doi: 10.2514/1.G003815.
[92] J. Hikes, A. J. Liounis, and J. A. Christian, "Parametric covariance model for horizon-based optical navigation," *J. Guid., Control, Dyn.*, vol. 40, no. 1, pp. 170–178, Jan. 2017, doi: 10.2514/1.G000708.
[93] E. J. Lefferts, F. L. Markley, and M. D. Shuster, "Kalman filtering for spacecraft attitude estimation," *J. Guid., Control, Dyn.*, vol. 5, no. 5, pp. 417–429, Sep. 1982, doi: 10.2514/3.56190.
[94] F. L. Markley and J. L. Crassidis, *Fundamentals of Spacecraft Attitude Determination and Control*. New York, NY, USA: Springer, 2014.
[95] D. Modenini, "Attitude determination from ellipsoid observations: A modified orthogonal Procrustes problem," *J. Guid., Control, Dyn.*, vol. 41, no. 10, pp. 2320–2325, 2018, doi: 10.2514/1.G003610.
[96] G. Wahba, "A least squares estimate of satellite attitude," *SIAM Rev.*, vol. 7, no. 3, p. 409, 1965, doi: 10.1137/1007077.
[97] F. L. Markley, "Attitude determination using vector observations and the singular value decomposition," *J. Astron. Sci.*, vol. 36, no. 3, pp. 245–258, 1988.
[98] B. F. Green, "The orthogonal approximation of an oblique structure in factor analysis," *Psychometrika*, vol. 17, no. 4, pp. 429–440, Dec. 1952, doi: 10.1007/BF02288918.
[99] P. H. Schönemann, "A generalized solution of the orthogonal Procrustes problem," *Psychometrika*, vol. 31, no. 1, pp. 1–10, Mar. 1966, doi: 10.1007/BF02289451.
[100] J. C. Gower and G. B. Dijksterhuis, *Procrustes Problems*. Oxford, U.K.: Oxford Univ. Press, 2004.
[101] P. H. Schönemann, "On two-sided orthogonal Procrustes problems," *Psychometrika*, vol. 33, no. 1, pp. 19–33, Mar. 1968, doi: 10.1007/BF02289673.
[102] T. Werner and T. Pajdla, "Cheirality in epipolar geometry," in *Proc. 8th IEEE Int. Conf. Comput. Vis. (ICCV)*, Jul. 2001, pp. 548–553, doi: 10.1109/ICCV.2001.937564.
[103] P. C. Thomas, "Sizes, shapes, and derived properties of the saturnian satellites after the Cassini nominal mission," *Icarus*, vol. 208, no. 1, pp. 395–401, Jul. 2010, doi: 10.1016/j.icarus.2010.01.025.
[104] D. Modenini, "Five-degree-of-freedom pose estimation from an imaged ellipsoid of revolution," *J. Spacecraft Rockets*, vol. 56, no. 3, pp. 952–958, May 2019, doi: 10.2514/1.A34340.
[105] G. H. Golub, "Some modified matrix eigenvalue problems," *SIAM Rev.*, vol. 15, no. 2, pp. 318–334, Apr. 1973, doi: 10.1137/1015032.

JOHN A. CHRISTIAN received the B.S. and M.S. degrees in aerospace engineering from the Georgia Institute of Technology and the Ph.D. degree in aerospace engineering from The University of Texas at Austin. From 2010 to 2012, he was an Engineer with the GNC Autonomous Flight Systems Branch, NASA Johnson Space Center. From 2013 to 2017, he was an Assistant Professor with West Virginia University. Since 2017, he has been on the faculty with the Rensselaer Polytechnic Institute, where he is currently an Associate Professor with the Department of Mechanical, Aerospace, and Nuclear Engineering. He has contributed to many past spaceflight missions. He is the author or a coauthor of more than 100 publications. His research program focuses on spacecraft navigation, computer vision, and planetary science. He is also an Associate Fellow of AIAA. He was a recipient of the NASA Innovative Advanced Concepts (NIAC) Fellow award and the AFOSR Young Investigator Program (YIP) award.