IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 3, 2021, accepted January 23, 2021, date of publication January 26, 2021, date of current version February 16, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3054836

Star Identification and Attitude Determination With Projective Cameras

JOHN A. CHRISTIAN[1] AND JOHN L. CRASSIDIS[2]
[1]Department of Mechanical, Aerospace, and Nuclear Engineering, Rensselaer Polytechnic Institute, Troy, NY 12180, USA
[2]Department of Mechanical and Aerospace Engineering, University at Buffalo, The State University of New York, Buffalo, NY 14260, USA

Corresponding author: John A. Christian (chrisj9@rpi.edu)

This work was supported in part by the U.S. National Aeronautics and Space Administration (NASA) under Award 80NSSC17M0027 and Award 80NSSC20K1018.

ABSTRACT Images of starfields collected by a projective camera are useful for a variety of scientific and engineering purposes. This utility is exemplified by star trackers, which are amongst the most commonly used sensors for determining the attitude of modern spacecraft. While the literature on star identification and star-based attitude determination is extensive, most algorithms are developed in an *ad hoc* manner. This work provides a comprehensive and systematic framework for invariant-based star identification and shows most past star identification algorithms to be special cases within this framework. The new star identification framework is found to motivate new problems in attitude determination and sensor self-calibration. Specifically, new algorithms are presented for simultaneous attitude determination and camera calibration for a generic wide field-of-view sensor using a single starfield image. In the special case where camera focal length is the only unknown calibration parameter, attitude determination performance of the new algorithm is indiscernible from a perfectly calibrated camera.

INDEX TERMS Asterisms, attitude determination, invariant theory, pattern recognition, spacecraft navigation, star identification, star trackers, total least squares, Wahba's problem.

I. INTRODUCTION

There are a variety of situations where it is necessary to autonomously recognize an asterism (i.e., a star pattern) in a digital image. This occurs when solving the lost-in-space attitude determination problem with star trackers [1], [2], when registering scientific astronomical images [3], and in other applications. The present manuscript is motivated by the spacecraft attitude determination application, which has seen numerous star identification algorithms proposed since the 1970s [4], [5].

For an asterism to be recognizable in a single image, there must be some attribute of the pattern that is always recoverable from only its appearance in the image. This attribute could be related to geometric or photometric properties of the asterism. While both are used, difficulty with accurately measuring the photometric properties of stars with low-cost sensors make it desirable to recognize asterisms with the pattern geometry alone. The usual approach is to find some descriptive attributes of the pattern that are *invariant* to attitude (and, perhaps, camera calibration) so that they may be computed ahead of time and stored in an onboard catalog. Then, when an image is collected aboard a spacecraft at an unknown attitude (and, perhaps, with an unknown calibration), these descriptive asterism attributes may be computed from the image and directly compared against the precomputed catalog.

There have been a great variety of asterism descriptors proposed over the years. These descriptors, however, are not as different from one another as they may first seem. Moreover, in many cases, the different descriptors solve slightly different pattern recognition problems—often without a clear discussion of this important fact. Therefore, following the philosophical approach of Refs. [6] and [7], this work provides a theoretical framework for developing invariant descriptors for asterisms as seen in an image from a projective camera. Popular existing asterism descriptors used for star identification are shown to be example cases within this framework.

Analysis of the geometry reveals that there are (at least) four fundamental classes of optical system for which invariant asterism descriptors may be built: generic calibrated camera,

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

narrow field-of-view (FOV) calibrated camera, generic uncalibrated camera, narrow FOV uncalibrated camera. Good descriptor options already exist for some of these cases and this manuscript describes novel descriptors for others. Of note, there have been other attempts to construct attitude invariant descriptors for generic (wide FOV) uncalibrated cameras (e.g., [8], [9])—but, as will be shown, these are not formally invariant for the generic problem.

Invariant asterism descriptors are the mathematical construct around which practical star identification algorithms are built. By themselves, these descriptors are analytically exact and numerical simulations provide only modest additional insight. However, to match an observed descriptor to its corresponding catalog entry in practice requires us to consider the interplay between (1) the stability of these descriptors in the presence of measurement noise, (2) the makeup of the full star catalog along with a scheme for choosing which star asterisms to index, and (3) the data structure used to mechanize the real-time index search on a digital computer. A specific star identification algorithm represents a specific choice for each of these three tasks. Numerical simulations and live-sky experiments between different algorithms are important here, though simple changes to any choice in the pipeline can significantly affect overall performance—often making a fair comparison between competing algorithms difficult. Numerical comparisons of some existing algorithms may be found elsewhere [4], [5], [10]. The contribution of this manuscript is not the development of new star identification algorithms, but a better theoretical framework for understanding how the vast majority of these algorithms actually function. Such a framework provides a valuable tool for spacecraft navigators that seek a more mathematically rigorous (as opposed to a heuristic) approach for developing and evaluating star identification algorithms.

Once observed stars have been matched to a catalog of known stars, many good algorithms exist for attitude determination with a calibrated camera [11]–[13]. These algorithms may usually be posed as a solution to Wahba's Problem [14]. When the camera is poorly calibrated, there are some algorithms for narrow FOV cameras (e.g., [15]) and some for wide FOV cameras (e.g., [9])—but these both assume some knowledge (though it may be poor) of the focal length. These algorithms commonly assume square pixels and that the other three calibration terms (detector skewness and coordinates of the principal point) are known. This work relaxes all of these assumptions and presents a novel method for attitude determination with a generic uncalibrated camera (regardless of FOV). Further, in the case when only the focal length is unknown (or, more precisely, when the ratio of focal length to pixel pitch $d_x = f/\mu_x$ is unknown), the general case collapses to an elegant and simple-to-implement solution. While the contribution of this manuscript in the area of star identification is a theoretical framework, the contribution in the area of attitude determination and self-calibration is a set of novel algorithms. The performance of these algorithms are verified with numerical simulations and compared to some of the most popular algorithms in use today.

II. GEOMETRY OF STAR OBSERVATIONS WITH A PROJECTIVE CAMERA

Most modern star catalogs (e.g., Hipparcos [16], [17], Gaia [18], [19]) describe the direction to star by a pair of angles (e.g., right ascension and declination) in the International Celestial Reference Frame (ICRF) [20]–[22]. These pairs of angles may also be interpreted as a unit vector,

$$\boldsymbol{e}_i = \begin{bmatrix} \cos\delta_i \cos\alpha_i \\ \cos\delta_i \sin\alpha_i \\ \sin\delta_i \end{bmatrix} \tag{1}$$

where α_i is the right ascension and δ_i is the declination. Effects such as stellar aberration, parallax, and proper motion [23], [24] are neglected for simplicity of the present discussion, though they may be added without any effect on the subsequent discussions. Note that the unit vector $\boldsymbol{e}_i$ represents a direction (a line) passing through the origin and, therefore, is described by a point in $\mathbb{P}^2$. The reader interested in more details on the properties of projective space (e.g., $\mathbb{P}^2$, $\mathbb{P}^3$, $\mathbb{P}^n$) is directed to Ref. [25] and Ref. [26].

Now, consider a projective camera that observes a set of stars. Let $\boldsymbol{T}$ be the attitude transformation matrix (passive interpretation of a rotation [27]) that transforms the ICRF star direction $\boldsymbol{e}_i$ into the same star direction as expressed in the camera sensor frame $\boldsymbol{a}_i$,

$$\boldsymbol{a}_i = \boldsymbol{T}\boldsymbol{e}_i \tag{2}$$

The reader is reminded that $\boldsymbol{T}$ is a 3×3 proper orthogonal matrix [$\det(\boldsymbol{T}) = +1$ and $\boldsymbol{T}^T\boldsymbol{T} = \boldsymbol{I}_{3\times3}$], which will be of importance later. Like $\boldsymbol{e}_i$, the unit vector $\boldsymbol{a}_i$ describes a point in $\mathbb{P}^2$.

An ideal projective camera is described by the pinhole camera model (see Fig. 1). Assuming the camera's z-axis is along the optical axis and is positive going out of the camera, the location $\boldsymbol{x} = [x_i, y_i]^T$ where the direction $\boldsymbol{a}_i = [X_i, Y_i, Z_i]^T$ pierces the image plane (a fictitious plane in front of the camera's optical center) may be found by similar triangles,

$$x_i = \frac{x_i}{1} = \frac{X_i}{Z_i} \quad \text{and} \quad y_i = \frac{y_i}{1} = \frac{Y_i}{Z_i} \tag{3}$$

When the image plane is placed at unit depth (as it is here), the resulting image plane coordinates are sometimes referred to as *focal-length normalized coordinates*.

It is usually simpler to work in homogeneous coordinates. In projective space (i.e., in $\mathbb{P}^2$), it is observed that $\boldsymbol{a}_i$ and $k\boldsymbol{a}_i$ represent the same point. Therefore, since $\boldsymbol{a}_i$ represents the line connecting the camera's center (i.e., origin of camera sensor frame and apparent pinhole location) to the star, it pierces the image plane (i.e., the $z = 1$ plane) at $\bar{\boldsymbol{x}}_i = [x_i, y_i, 1]^T$. Consequently, written in terms of projective geometry, the

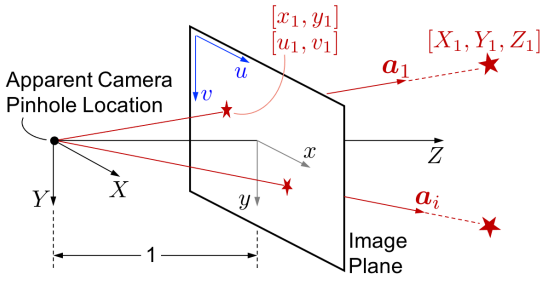

FIGURE 1. Illustration of camera frame, an image plane at unit depth, and star projection geometry with the pinhole camera model (perspective projection).

pinhole camera model is nothing more than

$$\bar{\boldsymbol{x}}_i = \frac{\boldsymbol{a}_i}{\boldsymbol{k}^T \boldsymbol{a}_i} \propto \boldsymbol{a}_i \tag{4}$$

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

The star observations are generally recorded in a 2D digital image as captured by a detector on the camera's focal plane. This may equivalently be described by discretization on the image plane. Using the conventions of [28], let the projective camera's $+z$ axis be along the optical axis (camera boresight) and pointing out of the camera. When looking out the camera (at an image) let the $+x$ axis be to the right (increasing column count in image) and the $+y$ axis be down (increasing row count in image). Therefore, if pixel coordinates are counted from the upper left-hand corner of the image, define the camera calibration matrix as

$$\boldsymbol{K} = \begin{bmatrix} d_x & \alpha & u_p \\ 0 & d_y & v_p \\ 0 & 0 & 1 \end{bmatrix} \tag{5}$$

where $d_x = f/\mu_x$ is the ratio of camera focal length to pixel pitch in the x direction (d_y is the same thing in the y direction), α is the detector skewness, and $[u_p, v_p]$ is the pixel location where the optical axis intersects the image (usually near the center of the image). The camera calibration matrix is nothing more than a change of coordinates (an affine transformation). Specifically, it transforms from units of length into image pixel coordinates. That is,

$$\begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} = \begin{bmatrix} d_x & \alpha & u_p \\ 0 & d_y & v_p \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = \boldsymbol{K}\bar{\boldsymbol{x}}_i \tag{6}$$

where u_i is the pixel column number and v_i is the pixel row number. Integer values of $[u_i, v_i]$ correspond to pixel centers.

Therefore, letting $\bar{\boldsymbol{u}}_i = [u_i, v_i, 1]^T$ and substituting Eq. (4) into Eq. (6), we quickly arrive at the result

$$\bar{\boldsymbol{u}}_i \propto \boldsymbol{K}\boldsymbol{a}_i \tag{7}$$

and, since $\boldsymbol{a}_i = \boldsymbol{T}\boldsymbol{e}_i$ from Eq. (2),

$$\bar{\boldsymbol{u}}_i \propto \boldsymbol{K}\boldsymbol{T}\boldsymbol{e}_i \tag{8}$$

The matrix $\boldsymbol{KT}$ is a 3×3 matrix with eight independent parameters (five from $\boldsymbol{K}$ and three from $\boldsymbol{T}$). It now becomes

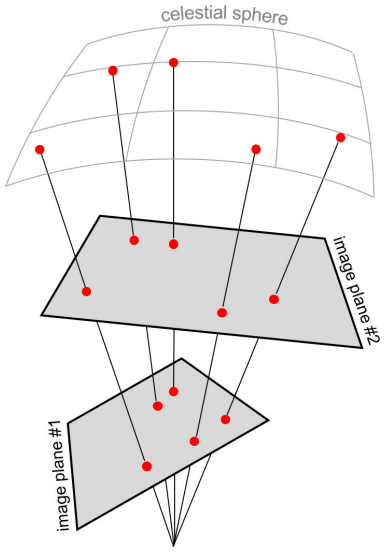

FIGURE 2. The same five stars (points on the celestial sphere) as viewed by two different camera image planes.

clear that Eq. (8) represents a homography. Therefore, defining the homography matrix $\boldsymbol{H}$ as

$$\boldsymbol{H} \propto \boldsymbol{K}\boldsymbol{T} \tag{9}$$

we arrive at

$$\bar{\boldsymbol{u}}_i \propto \boldsymbol{H}\boldsymbol{e}_i \tag{10}$$

It is observed, therefore, that the homography matrix $\boldsymbol{H}$ is only defined to an unknown scale—specifically, it is a 3×3 matrix having nine elements, but with only eight degrees of freedom. This interpretation of the star imaging problem (Fig. 2) allows for the straightforward deployment of results from algebraic geometry to the star identification and attitude determination problem.

III. ASTERISM DESCRIPTORS

To recognize an asterism (i.e., star pattern) using only the pixel coordinates of the stars in an image, we require a means of describing the pattern geometry that remains unchanged under the action of a projective camera at unknown attitude. This is achieved by computing algebraic quantities that are (1) functions of only the observed star coordinates, (2) invariant under changes in camera attitude, and (3) have different numerical values for each asterism. The collection of these algebraic quantities (referred to here simply as *invariants*) may be used to construct asterism descriptors. The proper mathematical framework for addressing this problem is invariant theory [29], [30].

Constructing invariants requires the consideration of both the algebraic variety and the group acting on that algebraic variety. From Eq. (1), it is observed that algebraic variety describing a star direction $\boldsymbol{e}_i$ is simply a point in $\mathbb{P}^2$. Therefore, letting $\mathscr{V}$ be the variety of a d-tuple of star directions, we find that $\mathscr{V} \cong (\mathbb{P}^2)^d$ has dimension $2d$.

Different assumptions about the camera give rise to different group actions on $\mathcal{Y}$. These different group actions, in turn, give rise to different invariants and different asterism descriptors. What follows is a discussion of the most common scenarios and the some of the most straightforward descriptors that can be developed for each.

A. GENERIC CALIBRATED CAMERAS

For a generic well-calibrated camera (either wide or narrow FOV), it is simple to recover the star directions in the camera frame, each of which is a point in $\mathbb{P}^2$. These star directions may be represented as unit vectors or, equivalently, as points on the celestial (unit) sphere. To make this explicit, the required computations are now briefly reviewed.

Given a star observed at pixel location $[u_i, v_i]$, we find the corresponding image plane coordinates $[x_i, y_i]$ by inverting Eq. (6) using the known calibration matrix $\boldsymbol{K}$,

$$\bar{\boldsymbol{x}}_i = \boldsymbol{K}^{-1}\bar{\boldsymbol{u}}_i \tag{11}$$

Making note of the structure of $\boldsymbol{K}$ in Eq. (5), it is possible to efficiently compute $\boldsymbol{K}^{-1}$ using the analytic inverse [31]

$$\boldsymbol{K}^{-1} = \begin{bmatrix} \frac{1}{d_x} & \frac{-\alpha}{d_x d_y} & \frac{\alpha v_p - d_y u_p}{d_x d_y} \\ 0 & \frac{1}{d_y} & \frac{-v_p}{d_y} \\ 0 & 0 & 1 \end{bmatrix} \tag{12}$$

Moreover, considering Eq. (4), Eq. (7), and the fact that $\boldsymbol{a}_i$ is a unit vector,

$$\boldsymbol{a}_i = \frac{\bar{\boldsymbol{x}}_i}{\|\bar{\boldsymbol{x}}_i\|} = \frac{\boldsymbol{K}^{-1}\bar{\boldsymbol{u}}_i}{\|\boldsymbol{K}^{-1}\bar{\boldsymbol{u}}_i\|} \tag{13}$$

Thus, it is simple to obtain a unit vector $\boldsymbol{a}_i$ from the observed star pixel coordinates $\bar{\boldsymbol{u}}_i = [u_i, v_i, 1]^T$ when the camera is calibrated (i.e., when $\boldsymbol{K}$ and $\boldsymbol{K}^{-1}$ are known).

When the measurements are unit vectors, the model and the observations are related to each other by the action of the three-dimensional special orthogonal group, SO(3). Thus, since a d-tuple of stars has dimension $2d$ and SO(3) has dimension 3, there are a total of $2d-3$ independent invariants for $d \geq 2$ stars in a calibrated camera image.

For a pair of stars ($d = 2$), there is one invariant ($4-3 = 1$). This single invariant is nothing more than the inter-star angle, which follows from the observation that the angle between two unit vectors is the same regardless of the frame in which the unit vectors are expressed. That is, using Eq. (2),

$$\cos\theta_{ij} = \boldsymbol{a}_i^T\boldsymbol{a}_j = \boldsymbol{e}_i^T\boldsymbol{T}^T\boldsymbol{T}\boldsymbol{e}_j = \boldsymbol{e}_i^T\boldsymbol{e}_j \tag{14}$$

where θ_{ij} is the inter-star angle between star i and star j. While not an especially deep geometric insight, the invariance of inter-star angle under the action of SO(3) is of immense practical use. Indeed, the inter-star angle is one of the most popular invariants used for describing an asterism. The inter-star angle is the fundamental invariant for some of the earliest published works on imaged-based start trackers for attitude determination [32], [33] and has since been used in a number of especially influential star identification pipelines [34], [35].

1) INVARIANTS FOR ASTERISMS OF THREE OR MORE STARS

For a triad of stars ($d = 3$), there are three independent invariants ($6 - 3 = 3$). Building on the invariant for a pair of stars, one solution is to use the three inter-star angles. The three inter-star angles are all independent for a triad. Note, however, that only a subset of the inter-star angles are independent for asterisms of four or more stars ($d \geq 4$); e.g., there are $\binom{4}{2} = 6$ possible inter-star angles for $d = 4$ stars, only five ($8 - 3 = 5$) of which are independent.

For an asterism of $d \geq 3$ stars, there exist other algebraic (or geometric) quantities that are invariant under the action of SO(3)—though these other invariants are necessarily *dependent* on the inter-star angles. This must be the case because we may always find $2d-3$ independent inter-star angles and there only exist $2d-3$ independent invariants (thus, there are no remaining independent invariants to be had). These other invariants are sometimes useful, since they may be more numerically stable in the presence of measurement noise or camera calibration error.

As an example, one alternative invariant for a triad of stars is to use the three dihedral angles of the spherical triangle (e.g., as is suggested in Refs. [9] and [34]). This is illustrated in Fig. 3. There are a few ways to compute the dihedral angles of a star triad using measurements from a calibrated camera or data from a star catalog. One way, which makes explicit the relaionship between the dihedral angles and inter-star angles, is to apply the spherical law of cosines to find

$$\cos\Theta_i = \frac{\cos\theta_{jk} - \cos\theta_{ij}\,\cos\theta_{ki}}{\sin\theta_{ij}\,\sin\theta_{ki}} \tag{15}$$

$$\cos\Theta_j = \frac{\cos\theta_{ki} - \cos\theta_{jk}\,\cos\theta_{ij}}{\sin\theta_{jk}\,\sin\theta_{ij}} \tag{16}$$

$$\cos\Theta_k = \frac{\cos\theta_{ij} - \cos\theta_{ki}\,\cos\theta_{jk}}{\sin\theta_{ki}\,\sin\theta_{jk}} \tag{17}$$

Since $\theta_{ij}, \theta_{jk}, \theta_{ki}$ are invariant, it follows that $\Theta_i, \Theta_j, \Theta_k$ are also invariant. Note, however, that while $\Theta_i, \Theta_j, \Theta_k$ form a complete set of invariants and are independent from each other, they are rational functions of (and, therefore, dependent on) $\theta_{ij}, \theta_{jk}, \theta_{ki}$. In either case—$\theta_{ij}, \theta_{jk}, \theta_{ki}$ or $\Theta_i, \Theta_j, \Theta_k$—there are only three independent invariants. The different sets of invariants are nothing more than different numeric representations of the same geometry. Moreover, both representations contain the exact same amount of information about the underlying asterism geometry.

A more direct way to compute the dihedral angles is to first recognize that the normal to the plane spanned by observations $\boldsymbol{a}_i$ and $\boldsymbol{a}_j$ is in the direction of $\boldsymbol{a}_i \times \boldsymbol{a}_j$. Therefore, the dihedral angles may be computed directly from the measurements or the catalog according to

$$\cos\Theta_i = \frac{(\boldsymbol{a}_i \times \boldsymbol{a}_j)^T\,(\boldsymbol{a}_i \times \boldsymbol{a}_k)}{\|\boldsymbol{a}_i \times \boldsymbol{a}_j\|\,\|\boldsymbol{a}_i \times \boldsymbol{a}_k\|} = \frac{(\boldsymbol{e}_i \times \boldsymbol{e}_j)^T\,(\boldsymbol{e}_i \times \boldsymbol{e}_k)}{\|\boldsymbol{e}_i \times \boldsymbol{e}_j\|\,\|\boldsymbol{e}_i \times \boldsymbol{e}_k\|} \tag{18}$$

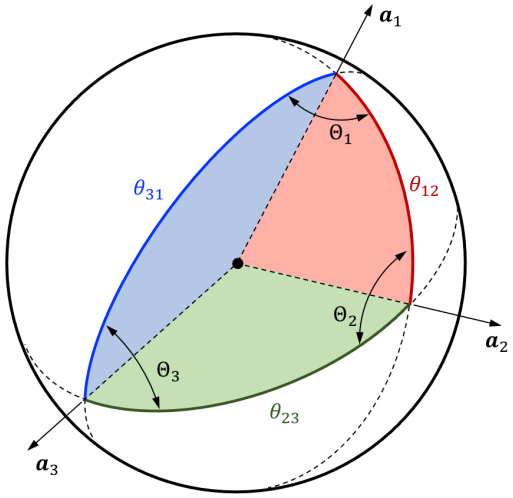

FIGURE 3. Illustration of spherical geometry showing inter-star angles ($\theta_{12}, \theta_{23}, \theta_{31}$) and dihedral angles ($\Theta_1, \Theta_2, \Theta_3$).

$$\cos\Theta_j = \frac{(\boldsymbol{a}_j \times \boldsymbol{a}_k)^T (\boldsymbol{a}_j \times \boldsymbol{a}_i)}{\|\boldsymbol{a}_j \times \boldsymbol{a}_k\| \, \|\boldsymbol{a}_j \times \boldsymbol{a}_i\|} = \frac{(\boldsymbol{e}_j \times \boldsymbol{e}_k)^T (\boldsymbol{e}_j \times \boldsymbol{e}_i)}{\|\boldsymbol{e}_j \times \boldsymbol{e}_k\| \, \|\boldsymbol{e}_j \times \boldsymbol{e}_i\|} \tag{19}$$

$$\cos\Theta_k = \frac{(\boldsymbol{a}_k \times \boldsymbol{a}_i)^T (\boldsymbol{a}_k \times \boldsymbol{a}_j)}{\|\boldsymbol{a}_k \times \boldsymbol{a}_i\| \, \|\boldsymbol{a}_k \times \boldsymbol{a}_j\|} = \frac{(\boldsymbol{e}_k \times \boldsymbol{e}_i)^T (\boldsymbol{e}_k \times \boldsymbol{e}_j)}{\|\boldsymbol{e}_k \times \boldsymbol{e}_i\| \, \|\boldsymbol{e}_k \times \boldsymbol{e}_j\|} \tag{20}$$

We clearly obtain the exact same values for $\Theta_i, \Theta_j, \Theta_k$ for any attitude $\boldsymbol{T}$, again showing that these quantities are invariant under the action of SO(3).

Finally, it is possible to mix the invariants $\theta_{ij}, \theta_{jk}, \theta_{ki}$ and $\Theta_i, \Theta_j, \Theta_k$ so long as no more than three are used. For example, Liebe [34] used an asterism descriptor comprised of the three independent invariants $\theta_{ij}, \theta_{ki}, \Theta_i$. It is also possible to choose a different number of stars and construct larger sets of independent invariants. For example, Toloei, *et al.*, [10] construct a descriptor using combinations of inter-star angles and dihedral angles amongst a set of five stars. However, although three different five-star descriptors are presented in Ref. [10], none of their proposed descriptors have the seven $(10 - 3 = 7)$ independent invariants expected for a five-star asterism. This is noteworthy since five-star descriptors of less than seven invariants leave helpful information unused, while five-star descriptors of more than seven invariants include redundant information that does not enhance descriptiveness. This highlights the importance of the insights gained by a rigorous accounting of the independent invariants that exist for any given scenario.

2) PERMUTATION AND PROJECTIVE INVARIANTS FOR A CALIBRATED CAMERA

The invariant sets $\theta_{ij}, \theta_{jk}, \theta_{ki}$ and $\Theta_i, \Theta_j, \Theta_k$ introduced above are permutation dependent. With only three stars, the observation indices may always be arranged in a consistent direction (clockwise or counterclockwise) such that only three options exist. That is, given three stars, the indices may be cyclically permuted to obtain (1,2,3), (2,3,1), or (3,1,2). The invariant descriptors of inter-star angle or dihedral angle will give different numerical values for each permutation. This is not inherently bad, since star identification and attitude determination ultimately require disambiguation of the star indices—however, it may not be desirable to search the catalog three times (once for each permutation) or to triple the size of the catalog (so it contains the values for each permutation).

One solution to this problem is to write asterism descriptors that are invariant to the ordering of the indices. There are a few ways this may be done. One option, as suggested by Cole and Crassidis [36], is to use the three unit vectors $\boldsymbol{a}_1, \boldsymbol{a}_2, \boldsymbol{a}_3$ (or $\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3$) to construct the area and polar moment of the triangle formed by their tips. These two quantities are clearly independent of the ordering of the unit vectors. There is, however, an additional (third) invariant that is possible since three unit vectors have three invariants under the action of SO(3).

Another option is to employ the method of Christian *et al.* [30], where the cyclic permutation of the indices is interpreted as the action of the group $\mathbb{Z}/3\mathbb{Z}$. This permits three ordered invariants—which applies to both $(\theta_{ij}, \theta_{jk}, \theta_{ki})$ and $(\Theta_i, \Theta_j, \Theta_k)$—to be cyclically permuted without changing the invariant descriptor. While the interested reader is directed to Ref. [30] for the derivation, the final result to make the 3-tuple (x, y, z) permutation invariant is

$$F_1(x, y, z) = x + y + z \tag{21}$$

$$F_2(x, y, z) = \gamma \Big[2(x^3 + y^3 + z^3) + 12xyz - 3(x^2y + y^2x + y^2z + z^2y + z^2x + x^2z)\Big] \tag{22}$$

$$F_3(x, y, z) = -3\sqrt{3}\gamma(x - y)(y - z)(z - x) \tag{23}$$

where

$$\gamma = \Big[x^2 + y^2 + z^2 - (xy + yz + zx)\Big]^{-1} \tag{24}$$

The reader may easily verify that this produces the same (F_1, F_2, F_3) for (x, y, z), (y, z, x), and (z, x, y). To our knowledge, Eqs. (21)–(23) have not been applied to the star identification problem before.

Although permutation invariants allow for matching an observed asterism to the catalog without considering the specific index assignments within that asterism, disambiguation of the indices must still occur to verify a hypothesized observation-to-catalog match.

3) REMARKS ON POORLY CALIBRATED CAMERA

In the case of a poorly calibrated camera, we may choose to either (1) exploit what calibration information may be available or (2) treat the camera as completely uncalibrated. This section focuses exclusively on the former, while the latter is discussed in Section III-C.

For a poorly calibrated camera, we may still convert the pixel coordinates of star centroids in an image into their corresponding unit vectors using Eq. (13)—though these unit vectors may be of lower quality when compared to a well-calibrated camera. Nonetheless, all of the results from

Section III-A still apply here and we find there to be $2d-3$ invariants described by inter-star angles, dihedral angles, or some other quantity that is an algebraic function of these invariant angles. The fundamental challenge here is the *stability* of these invariants under perturbations to elements within the camera calibration matrix $\boldsymbol{K}$.

The poorly calibrated camera scenario most plausibly occurs within the context of space exploration when an earlier calibration (e.g., a pre-flight calibration) is degraded due to environmental conditions (e.g., launch vibrations, thermal strain). As discussed in Ref. [28], it is standard practice to fix the principal point coordinates $[u_p, v_p]$. This is done since small errors in $[u_p, v_p]$ are indistinguishable from small errors in sensor alignment. Assuming the pixels remain square (they usually do), we also find $d_x = d_y$ and $\alpha = 0$. Thus, the ratio of focal length to pixel pitch ($d_x = f/\mu_x$) is the most important term to account for when considering an old and degraded calibration. It is advisable to work on d_x instead of f and μ_x since neither f nor μ_x are observable in practice (and it is only the ratio d_x that matters).

It was suggested by Leake, Arnas, and Mortari in Ref. [9] that the star triad dihedral angles $\Theta_i, \Theta_j, \Theta_k$ from Eqs. (15)–(17) are less sensitive to errors in focal length than the inter-star angles. This is generally true, though two points of clarification are required. First, the sensitivity is dependent on the ratio $d_x = f/\mu_x$ and not simply focal length, as perturbations in the assumed value of the pixel pitch μ_x alter the solution just as much as perturbations in the focal length f. Second, while the dihedral angles are indeed less sensitive to variations $d_x = f/\mu_x$, they are often *more* sensitive to errors in measured star location. This is easily demonstrated through a numerical simulation. Consider, for example, a camera with a 20 deg FOV and a $2,048 \times 2,048$ pixel FPA—corresponding to a camera having $d_x = 5,807.4$. Now, using this camera, suppose we view a random section of sky populated with stars from the Hipparcos star catalog [16], [17]. This produces a synthetic image with the apparent star pattern shown in Fig. 4. Let us now construct an example star triad using the stars circled in red. We consider three cases using this example star triad:

1) Perfect knowledge of d_x and a star centroiding error of 0.1 pixel.
2) An error of 0.1% in d_x and perfect knowledge of star centroid locations.
3) An error of 0.1% in d_x and a star centroiding error of 0.1 pixel.

A 10,000-run Monte Carlo simulation was performed for each of these cases, with the results summarized in Table 1. We clearly see that the inter-star angles are more stable in Case 1 (when errors in star centroiding dominate) and that the dihedral angles are more stable in Case 2 (when errors in d_x dominate). Case 3 shows that it is possible for the stability of inter-star angles and dihedral angles to be comparable when there is uncertainty in both measured star centroids and in d_x, though oftentimes one or the other dominates in practice.

TABLE 1. Numerical simulation on the stability of star triad invariants. Results are for a 10,000-run Monte Carlo study using the three example stars circled in red from Fig. 4.

Invariant Angle	Mean Value (deg)	Standard Deviation (arcsec) Case 1	Case 2	Case 3
θ_{ij}	10.18	4.79	35.00	35.21
θ_{jk}	10.31	5.07	35.79	35.96
θ_{ki}	12.93	4.77	44.79	44.97
Θ_i	51.64	30.40	2.76	30.46
Θ_j	78.51	37.57	3.02	37.77
Θ_k	50.75	29.14	0.47	29.14

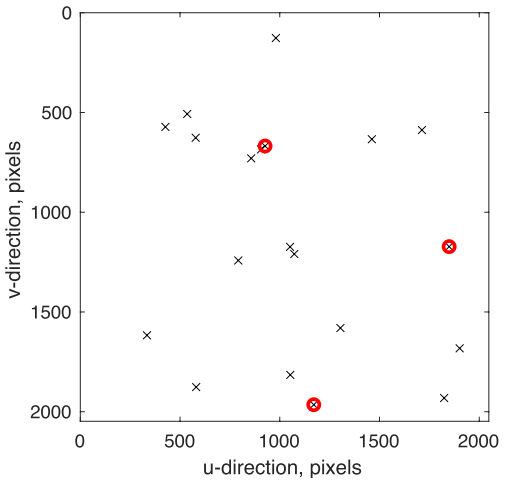

FIGURE 4. Illustration of synthetic star pattern from Hipparcos star catalog for a 20 deg FOV camera. All visible stars are show with a black ×. An example star triad is highlighted with red circles.

These results are consistent with the numerical experiments reported in Ref. [9], though our analysis provides a more direct explanation of why certain cases demonstrate better matching performance than others.

B. CALIBRATED CAMERA WITH A NARROW FIELD-OF-VIEW (FOV)

When the camera FOV is narrow (a few degrees or less), the z-component of each star observation in the camera frame ($\boldsymbol{a}_i$) is nearly one. In this case, the apparent location of each star in an image may often be approximated by the so-called weak perspective projection model (i.e., scaled orthographic projection). If the camera is calibrated, then the scaling is known. Therefore, since the tangent plane to the celestial sphere is parallel to the image plane, the apparent asterism pattern only varies by a Euclidean transformation (translation and rotation). The reader is reminded that this approximation is never exactly true—even for a perfect pinhole camera.

Therefore, assuming a narrow FOV camera, asterisms are transformed by the action of the 2D Euclidean group, E(2). The 2D Euclidean group has dimension 3. Thus, since a d-tuple of stars has dimension $2d$, there are a total of $2d-3$ independent invariants for $d \geq 2$ stars in a calibrated narrow FOV image. For a pair of stars ($d = 2$) the unique invariant is

the distance in the image (the length) between the two stars. For three stars, there exist three $(6-3=3)$ invariants, with the most natural being the three inter-star distances in the image. The inter-star distances in an image have become a very popular invariant for constructing asterism descriptors, being used in a variety of star identification pipelines [37]. Another popular option is to align the pattern with two reference stars (e.g., one star as the origin with the other star defining the first coordinate direction) and use the transformed coordinates of the stars as the descriptor [38].

Observe now that there are $2d-3$ independent invariants for both the generic calibrated camera and the narrow FOV calibrated camera. Thus, no additional information (either exact or approximate) is gained by restricting the problem to a narrow FOV to allow for asterisms to be treated as planar point patterns that only undergo simple Euclidean transformations. Moreover, if the camera is well-enough calibrated to make the distances on the focal plane reliable, it is well-enough calibrated to produce the unit vectors $\{\boldsymbol{a}_i\}_{i=1}^{d}$. Invariants built on the unit vector are exact for a projective camera, whereas the distances are only approximately invariant. Considering these observations, we find no compelling reason to ever make the narrow FOV assumption (i.e., the assumption of star patterns only undergoing 2D Euclidean transformations in the image) the a when building an asterism descriptor.

C. GENERIC UNCALIBRATED CAMERAS

For a generic uncalibrated camera (either wide or narrow FOV), the projection of stars onto an image is governed by Eq. (10). That is, the transformation is described by the action of the projective general linear group, PGL(3), which has dimension 8. Thus, since a d-tuple of stars has dimension $2d$ and PGL(3) has dimension 8, there are a total of $2d-8$ independent invariants for $d \geq 5$ stars in an uncalibrated camera image.

There appears to be some confusion in the existing literature on the construction of asterism descriptors for an uncalibrated camera. A number of descriptors have been proposed, but few of them are formally invariant under PGL(3). The fundamental problem is repeated attempts to apply the invariants (or, modificaitons of the invariants) arising from action of SO(3) on unit vectors to the case of 2D images formed by the action of PGL(3). Unfortunately, these are different problems and their invariants are not interchangeable. Some of the most popular descriptors are now reviewed and are shown to not be invariant for an uncalibrated wide FOV camera under perturbations in either attitude or in camera calibration. This provides motivation for a different approach.

Based on the influential work of Samaan, Mortari, and Junkins [15], it has been widely suggested [8], [39] that the two independent interior angles of the apparent triangle in an image can be used to describe a star triad for an uncalibrated camera (see Fig. 5). It is immediately evident that these interior angles are not invariant, since no invariants exist for asterisms containing less than five stars. This may

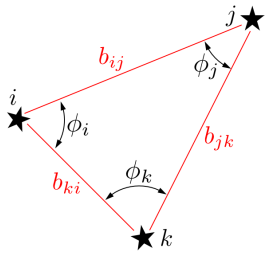

FIGURE 5. Illustration of interior angles for planar star triangle as it appears in an image. The interior angles (ϕ_i, ϕ_j, ϕ_k) and the triangle edge lengths (b_{ij}, b_{jk}, b_{ki}) are not invariant to general changes in attitude or camera calibration.

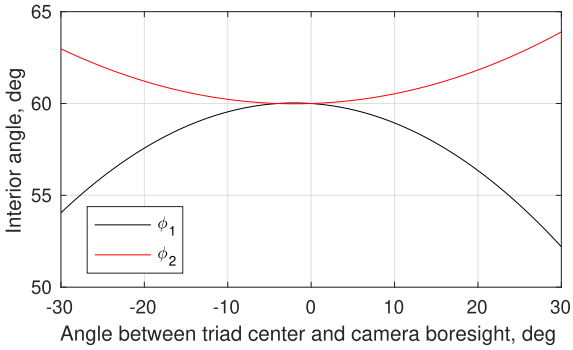

FIGURE 6. For a wide FOV camera, thee interior angles to a star triad change as the pattern moves across the image.

be verified by a simple numerical simulation. Consider a triad of stars forming an equilateral triangle on the celestial sphere, with each star being 2 deg away from the triangle center [and, therefore, the stars being about 3.46 deg apart from each other; i.e. $\theta_{ij} = 3.46$ deg using Eq. (14)]. When centered along the camera optical axis, this three-star astrism projects to an equilateral triangle in the image. Rotating the camera so the optical axis is an angle ϕ from the star triad center (positive in the direcion away from star i), we obtain the image triangle interior angles shown in Fig. 6. These angles are clearly not constant, thus the interior angles are not attitude invariant [nor are they invariant under PGL(3)]. Considering these drawbacks of existing asterism descriptors for uncalibrated cameras, there is a need to explore alternate approaches.

1) INVARIANTS FROM DETERMINANTS

The projective invariants for an asterism of five stars may be computed using the well-known invariant for five coplanar points [40]. Specifically, making use of Eq. (10), observe that

$$
\begin{aligned}
I_1 &= \frac{\det([\bar{\boldsymbol{u}}_2\ \bar{\boldsymbol{u}}_3\ \bar{\boldsymbol{u}}_1])\det([\bar{\boldsymbol{u}}_4\ \bar{\boldsymbol{u}}_5\ \bar{\boldsymbol{u}}_1])}{\det([\bar{\boldsymbol{u}}_2\ \bar{\boldsymbol{u}}_4\ \bar{\boldsymbol{u}}_1])\det([\bar{\boldsymbol{u}}_3\ \bar{\boldsymbol{u}}_5\ \bar{\boldsymbol{u}}_1])} \\
&= \frac{\det(\boldsymbol{H}[\boldsymbol{e}_2\ \boldsymbol{e}_3\ \boldsymbol{e}_1])\det(\boldsymbol{H}[\boldsymbol{e}_4\ \boldsymbol{e}_5\ \boldsymbol{e}_1])}{\det(\boldsymbol{H}[\boldsymbol{e}_2\ \boldsymbol{e}_4\ \boldsymbol{e}_1])\det(\boldsymbol{H}[\boldsymbol{e}_3\ \boldsymbol{e}_5\ \boldsymbol{e}_1])} \\
&= \frac{\det(\boldsymbol{H})^2}{\det(\boldsymbol{H})^2}\frac{\det([\boldsymbol{e}_2\ \boldsymbol{e}_3\ \boldsymbol{e}_1])\det([\boldsymbol{e}_4\ \boldsymbol{e}_5\ \boldsymbol{e}_1])}{\det([\boldsymbol{e}_2\ \boldsymbol{e}_4\ \boldsymbol{e}_1])\det([\boldsymbol{e}_3\ \boldsymbol{e}_5\ \boldsymbol{e}_1])}
\end{aligned}
$$

$$= \frac{\det([\boldsymbol{e}_2\ \boldsymbol{e}_3\ \boldsymbol{e}_1])\det([\boldsymbol{e}_4\ \boldsymbol{e}_5\ \boldsymbol{e}_1])}{\det([\boldsymbol{e}_2\ \boldsymbol{e}_4\ \boldsymbol{e}_1])\det([\boldsymbol{e}_3\ \boldsymbol{e}_5\ \boldsymbol{e}_1])} \tag{25}$$

Hence, I_1 may be computed from the observed star image coordinates $\{\bar{\boldsymbol{u}}_i\}_{i=1}^5$ or from the star catalog ICRF unit vectors $\{\boldsymbol{e}_i\}_{i=1}^5$ and the same value is always computed regardless of $\boldsymbol{H}$ (i.e., we compute the same value I_1 for any camera attitude and for any camera calibration parameters). There are two independent invariants of this form that may be computed as rational functions of the data (either observations or catalog). The determinant-based invariant above clearly holds for any reasonable permutation of the five points. For example, the following two point combinations produce independent invariants

$$\begin{aligned} I_1 &= \frac{\det([\bar{\boldsymbol{u}}_2\ \bar{\boldsymbol{u}}_3\ \bar{\boldsymbol{u}}_1])\det([\bar{\boldsymbol{u}}_4\ \bar{\boldsymbol{u}}_5\ \bar{\boldsymbol{u}}_1])}{\det([\bar{\boldsymbol{u}}_2\ \bar{\boldsymbol{u}}_4\ \bar{\boldsymbol{u}}_1])\det([\bar{\boldsymbol{u}}_3\ \bar{\boldsymbol{u}}_5\ \bar{\boldsymbol{u}}_1])} \\ &= \frac{\det([\boldsymbol{e}_2\ \boldsymbol{e}_3\ \boldsymbol{e}_1])\det([\boldsymbol{e}_4\ \boldsymbol{e}_5\ \boldsymbol{e}_1])}{\det([\boldsymbol{e}_2\ \boldsymbol{e}_4\ \boldsymbol{e}_1])\det([\boldsymbol{e}_3\ \boldsymbol{e}_5\ \boldsymbol{e}_1])} \end{aligned} \tag{26}$$

and

$$\begin{aligned} I_2 &= \frac{\det([\bar{\boldsymbol{u}}_1\ \bar{\boldsymbol{u}}_3\ \bar{\boldsymbol{u}}_2])\det([\bar{\boldsymbol{u}}_4\ \bar{\boldsymbol{u}}_5\ \bar{\boldsymbol{u}}_2])}{\det([\bar{\boldsymbol{u}}_1\ \bar{\boldsymbol{u}}_4\ \bar{\boldsymbol{u}}_2])\det([\bar{\boldsymbol{u}}_3\ \bar{\boldsymbol{u}}_5\ \bar{\boldsymbol{u}}_2])} \\ &= \frac{\det([\boldsymbol{e}_1\ \boldsymbol{e}_3\ \boldsymbol{e}_2])\det([\boldsymbol{e}_4\ \boldsymbol{e}_5\ \boldsymbol{e}_2])}{\det([\boldsymbol{e}_1\ \boldsymbol{e}_4\ \boldsymbol{e}_2])\det([\boldsymbol{e}_3\ \boldsymbol{e}_5\ \boldsymbol{e}_2])} \end{aligned} \tag{27}$$

Since I_1 and I_2 are independent and a pattern of five stars has only two independent invariants, it follows that any other projective invariant one may conceive for a five-star asterism must be dependent on I_1 and I_2. We may gain further insight by considering these invariants within the context of a cross ratio.

2) INVARIANTS FROM THE CROSS RATIO

It is well-known that four points along a line produce a projective invariant in form of the cross ratio [41]. Making use of the fact that points are dual to lines in $\mathbb{P}^2$ [25], it follows that four lines through a common point also form an invariant cross ratio.

Therefore, consider five points. Choose four points and then construct the four lines from these that pass through the fifth point. The cross-ratio of these lines is invariant under the action of PGL(3). To make this explicit, recall that in homogeneous coordinates the line $\boldsymbol{\ell}_{ij}$ formed by the join of points $\bar{\boldsymbol{u}}_i$ and $\bar{\boldsymbol{u}}_j$ is

$$\boldsymbol{\ell}_{ij} \propto \bar{\boldsymbol{u}}_i \times \bar{\boldsymbol{u}}_j \tag{28}$$

where a point $\bar{\boldsymbol{w}}$ on line $\boldsymbol{\ell}_{ij}$ satisfies the constratint $\boldsymbol{\ell}_{ij}^T\bar{\boldsymbol{w}} = 0$. Thus, choosing star $\bar{\boldsymbol{u}}_1$ as the reference point, we may compute the cross ratio $\mathrm{Cr}(\boldsymbol{\ell}_{12}, \boldsymbol{\ell}_{13}, \boldsymbol{\ell}_{14}, \boldsymbol{\ell}_{15})$ and see that it is constant. There are a variety of ways to compute the cross ratio, though the simplest is by use of determinants. In this case, it may ultimately be shown that

$$\mathrm{Cr}(\boldsymbol{\ell}_{12}, \boldsymbol{\ell}_{13}, \boldsymbol{\ell}_{14}, \boldsymbol{\ell}_{15}) = \frac{\det([\bar{\boldsymbol{u}}_2\ \bar{\boldsymbol{u}}_3\ \bar{\boldsymbol{u}}_1])\det([\bar{\boldsymbol{u}}_4\ \bar{\boldsymbol{u}}_5\ \bar{\boldsymbol{u}}_1])}{\det([\bar{\boldsymbol{u}}_2\ \bar{\boldsymbol{u}}_4\ \bar{\boldsymbol{u}}_1])\det([\bar{\boldsymbol{u}}_3\ \bar{\boldsymbol{u}}_5\ \bar{\boldsymbol{u}}_1])} \tag{29}$$

which is exactly the same as Eq. (26). Choosing a different point (other than $\bar{\boldsymbol{u}}_1$) as the reference will produce an independent invariant. Thus, we also note that the determinant-based invariants of Eqs. (26) and (27) may be viewed as particular cross-ratios.

3) INVARIANTS BY MAPPING TO A CANONICAL FRAME

An alternate approach is to choose four stars and to find the homography that maps them to a canonical frame; e.g., the unit square defined by the corner points [0,0], [0,1], [1,0], [1,1]. Four star observations in an image, with no three belonging to a common line, uniquely define this transformation. The resulting homography may then be used to map additional stars (e.g, a fifth or sixth star) into this same transformed space. Since the action of PGL(3) itself is defined by a homography, this procedure will always map the coordinates of the additional stars in an image to the same place in the transformed space—hence, the transformed coordinates of these "extra" stars may be interpreted as invariants of the pattern. This procedure was suggested in both Ref. [40] and Ref. [42] as an invariant for a pattern of points in $\mathbb{P}^2$, and has subsequently been used in other applications [43]. To our knowledge, this approach has not yet been applied to star identification.

4) PERMUTATION AND PROJECTIVE INVARIANTS FOR AN UNCALIBRATED CAMERA

The invariants for an asterism of five stars discussed so far are dependent on the ordering of the five stars. Moreover, unlike the case of three stars, it is not generally possible to use a simple clockwise/counterclockwise convention to distinguish between patterns that are mirror images of one another. Thus, each five-star asterism possesses a total of $5! = 120$ possible permutations, giving rise to 30 different invariant values (five sets of six, $5 \times 6 = 30$). One way to address this problem is to develop invariants that are insensitive to the arbitrary labeling of the stars in an observed asterism—this will produce a set of five possible invariant values, only two of which are independent.

Projective and permutation (p^2) invariants under PGL(3) for a set of five points in $\mathbb{P}^2$ has been studied extensively [44], [45]. The concept may be straightforwardly developed from the results of Section III-C2. Suppose we arbitrarily label the points with indices 1 to 5, choose point 1 as the reference point, and then compute a cross ratio $\tau = \mathrm{Cr}(\boldsymbol{\ell}_{12}, \boldsymbol{\ell}_{13}, \boldsymbol{\ell}_{14}, \boldsymbol{\ell}_{15})$ from Eq. (29). It is well known that the $4! = 24$ permutations of four lines (corresponding here to the lines we obtain from the 24 possible permutations of points 2,3,4,5) produce six different cross ratio values [41]

$$\tau,\ \tau^{-1},\ 1-\tau,\ (1-\tau)^{-1},\ \tau(\tau-1)^{-1},\ \tau^{-1}(\tau-1) \tag{30}$$

Moreover, there exists a rational function of the cross ratio—the so-called *j-invariant*—that is the same when any of the six cross ratio values from Eq. (30) are used. The classical

j-invariant is given by [46]

$$J(\tau) = \frac{(\tau^2 - \tau + 1)^3}{\tau^2(\tau - 1)^2} \tag{31}$$

which the reader may verify produces the same value J if any of the six cross ratios from Eq. (30) are used as the input argument. This, however, does not produce a bounded value for $J(\tau)$. Therefore, following Ref. [44], it is sometimes convenient to write the p^2 invariant

$$J'(\tau) = \frac{2\tau^6 - 6\tau^5 + 9\tau^4 - 8\tau^3 + 9\tau^2 - 6\tau + 2}{\tau^6 - 3\tau^5 + 3\tau^4 - \tau^3 + 3\tau^2 - 3\tau + 1} \tag{32}$$

which is both bounded ($J' : \mathbb{R} \rightarrow [2.0, 2.8]$) and permutation invariant. Once again, the reader may verify that the same value for J' is obtained if any of the six cross ratios from Eq. (30) are used as the input argument.

For a set of five points, it follows that we may compute five different j- or p^2 invariants from Eq. (31) or Eq. (32)—one using each of the five points as the reference point for forming the four lines used in the cross ratio. While five possible p^2-invariant values may be computed, only two of these five are independent.

The primary difficulty with the p^2 (or j-) invariants is which of the five values to use as the descriptor. The authors of Ref. [45] suggest to use all five p^2 invariants (sort them and place in a vector), though this approach is not space-optimal as it contains three dependent values (since only two of the five are independent). In principle, it is possible to pick any two of the five entries as the descriptor, but this does not provide robust noise performance due to the way the cross ratio values are mapped to $J(\tau)$ or $J'(\tau)$. A related phenomenon was also seen in the problem of crater identification in Ref. [30]. At this point, it remains unclear if p^2 invariants are a wise choice for creating asterism descriptors in practice. This is an area worthy of further study.

D. UNCALIBRATED CAMERAS WITH A NARROW FIELD-OF-VIEW (FOV)

When the camera FOV is narrow (a few degress or less), the image formation may be approximated by a scaled orthograpic projection (similar to Sec. III-B). However, unlike the case of the calibrated camera, the scaling is unknown when the camera is uncalibrated. Thus, instead of a Eucledian transformation, the apparent asterism varies by a similarity transformation (translation, rotation, scaling) in any given uncalibrated image from a narrow FOV camera. The reader is again reminded that this approximation is never exactly true—even for a perfect pinhole camera.

Therefore, for a narrow FOV camera, asterisms are transformed by the action of S(2) (2D similarity group). A 2D similarity transformation has dimension 4. Thus, since a d-tuple of stars has dimension $2d$, there are a total of $2d - 4$ independent invariants for $d \geq 2$ stars in an uncalibrated camera image. For a triad of stars there are two ($6 - 4 = 2$) independent invariants. The two independent interior angles of a triangle used in Ref. [15] are an example. It is really only in this special case where the proposed descriptor of Ref. [15] is strictly valid.

For a triad of stars with two independent invariants, there are other notable examples of descriptors beyond the influential work of Ref. [15] that are invariant to similarity transformations. Of note is the much earlier work of Groth [47] that describes a triad of stars using one interior star angle (ϕ_i from Fig. 5) and the ratio of two triangle edge lengths ($r_i = b_{ij}/b_{ki}$, with b_{ij} and b_{ij} from Fig. 5)

It is also observed that the popular four-star descriptor from Ref. [3] for the astrometric registration of scientific images also falls into this category—which is well justified given the very narrow FOV of telescopes used for astronomical research. In the case of Ref. [3], a similarity transformation is applied to a four-star asterism's image pattern such that the two stars furthest apart from one another map to the points [0, 0] and [1, 1]. The coordinates of the remaining two stars in this canonical frame are taken as the pattern's descriptor. This is conceptually identical to the approach discussed in Section III-C3, but the mapping to the canonical frame can now be computed with only two points since the similarity transform only has dimension 4. Further, since there are $2d - 4$ independent invaraints for point patterns under S(2), it is observed that this four-element descriptor is consistent with the $8 - 4 = 4$ invariants expected for a four-star asterism (i.e., a star quad).

IV. MATCHING ASTERISM DESCRIPTORS TO A STAR CATALOG

Regardless of the assumed camera type (generic calibrated, narrow FOV calibrated, generic uncalibrated, narrow FOV uncalibrated) or the specific invariants chosen for that camera type, we obtain a set of numerical values (the invariants) from the observed asterism that must be matched to corresponding values from a star catalog. This type of invariant-based indexing problem is not unique to star identification and occurs widely in other application domains [7], [42], [48], [49].

The fundamental idea behind invariant-based matching is that catalog asterisms and measured image asterisms are mapped into a common *index space* where direct comparison is always possible (see Fig. 7). For the case of star identification, each star in the catalog space and image space is a point in $\mathbb{P}^2$. Defining G to be the number of independent invariants (e.g., $G = 2d - 3$ for asterisms of d stars as viewed by a calibrated camera; see Table 2), each asterism may be mapped to a single point in $\mathbb{R}^G$. One simple way to describe this point in index space is with a $G \times 1$ vector formed by concatenation of the G independent invariants in a specified order. The vast majority of past star identification algorithms are simply different schemes to find the closest catalog points to a query point (formed by the observed asterism in an image) within the index space, though few of these algorithms make the abstraction to index space explicit.

When viewed using the index space abstraction, most successful star identification algorithms have the same overall framework. First, stars are grouped into asterisms (containing

TABLE 2. Number of independent invariants for an asterism of d stars.

Camera Type	Group Action	Number of Independent Invariants, G
General Calibrated Camera	Special Orthogonal Group, SO(3)	$2d-3$
Narrow FOV Calibrated Camera	Euclidean Group, E(2)	$2d-3$
General Uncalibrated Camera	Projective General Linear Group, PGL(3)	$2d-8$
Narrow FOV Uncalibrated Camera	Similarity Group, S(2)	$2d-4$

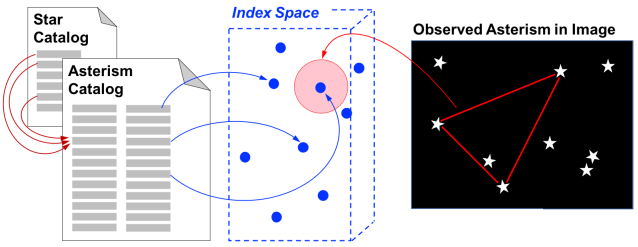

FIGURE 7. Mapping of entries from a catalog of known star asterisms and from a measured image asterism into a common index space. Measurement noise requires us to search a small region of the index space to find all potential matches from the catalog.

two or more stars) and invariants are computed using data from the star catalog and placed into an index. This task is performed offline and infrequently (perhaps only once). Later, an image is captured and the same invariants are computed. These invariants are used to query the index and produce a small list of feasible matches for the observed asterisms. These match hypotheses are then verified using additional information by additional checks.

While the conceptual framework is the same, practical challenges with implementing the above steps lead to a number of different (and equally reasonable) ways of performing robust star identification. The majority of these differences arise from how one chooses to balance the competing priorities of asterism uniqueness, index size, and index query speed. Different choices for the invariants discussed in Section III lead to even more variations on this common theme. It should come as no surprise, therefore, that so many different star identification algorithms have been published over the last 50 years [4], [5].

Asterisms containing more stars (large d) are more unique. This improving uniqueness occurs since each asterism is a point in index space and more stars increases the dimension of the index space (large $d \rightarrow$ large G). In general (not just for star identification), the number of feasible index matches for any given query decreases exponentially with an increase in G [48]. This would suggest forming asterisms of many stars, since increasing d would increase the likelihood that an observed asterism produces only one feasible match in the index. Moreover, for a sufficiently large G there is no need for subsequent match verification since the likelihood of a false match occurring (i.e., having a randomly occurring nearby point in index space) becomes vanishingly small in a very high dimensional index space.

The direct matching of asterisms containing many stars (large d) is often impractical. For a star catalog containing M stars, the worst case number of asterisms is $\binom{M}{d}$. Since it is always the case that $M \gg d$, the number of asterisms grows exponentially as d increases. Thus, using asterisms of more stars leads to an exponential growth in index size and the problem quickly becomes intractable.

The intractability of catalog size is almost always addressed using a two-part strategy. The first part is to not consider all possible $\binom{M}{d}$ asterisms. Presuming the star identification is to be performed on a digital image from a projective camera, the sensor FOV is limited and there is no need to consider asterisms whose angular extent is larger than the camera FOV. For example, consider two of the most popular contemporary star identification methods: the Pyramid approach [35] and the so-called "Astrometry.net" approach [3]. Pyramid creates an index of inter-star angles corresponding to two-star asterisms, with the index limited to only star pairs whose angular separation is less than a maximum threshold (usually taken as the camera FOV). Astrometry.net uses a similarity transformation to create a four-dimensional descriptor (what they call a "geometric hash code") corresponding to a four-star asterism (a star quad), with the index limited by tiling the celestial sphere using HEALPix [50]. HEALPix tiles are chosen to be about 1/3 the size of an image and only the brightest stars in each tile are kept. Then a specified number of four-star asterisms smaller than a specified size are generated for each HEALPix tile (using the brightest stars first). In both cases (Pyramid and Astrometry.net) the index building procedure produces substantially fewer than $\binom{M}{d}$ entries. Other star identification pipelines use other approaches, though the end goal managing index size while maintaining good coverage of possible asterisms is essentially the same.

The second part of the usual strategy for dealing with a large asterism index is to query the index using efficient data structures. While the practical constraints of the first part does slow the exponential growth of index size with increasing d, larger asterisms still produce a larger index—oftentimes too large to permit brute force matching. Fast matching to the index is essential since this task is often performed many times before a match is verified. If an image contains N observed stars, there are $\binom{N}{d}$ asterisms that can be built from these stars. With $d!$ permutations of star assignments for each asterism, there are a total of up to $d!\binom{N}{d} = N!/(N-d)!$ possibilities that must be considered. The average time to find a solution can sometimes be improved by not dwelling on

certain star observations (which may be false returns or excessively noisy) when checking these combinations and there are a number of deterministic pattern shifting algorithms for doing this [51].

There are a variety of data structures that may be used to accelerate real-time matching, including a k-d tree [52], R-tree [53], an n-d k-vector [54], or others [49], [55]. As an example, consider again the popular Pyramid and Astrometry.net pipelines. The original Pryramid approach [35] selects three observed stars and computes the three corresponding inter-star angles. For each inter-star angle, a separate index search is performed using 1-d k-vector [56], [57] to find feasible corresponding star pairs from the catalog. The results are combined to see if there is a unique set of star correspondences that amongst the index returns. Thus, Pyramid is an example of a case where searching is done one dimension at a time (rather than all at once)—though recent improvements have resulted in the n-d k-vector that would allow for a three-dimensional index query [54]. The Astrometry.net approach [3] uses a four-dimensional descriptor which is directly matched with a single query of a four-dimensional k-d tree. Some other star identification algorithms also use k-vectors (e.g., [15], [37], [58], [59]) or k-d trees (e.g., [60]), while others use completely different data structures.

The compromise between asterism uniqueness (which is better for a larger d) and index size (which is better for a smaller d) generally drives solutions star identification pipelines towards the smallest possible d. This usually requires match hypotheses be verified since the dimension of G is not high enough to keep the likelihood of a false match below an acceptable threshold. The Pyramid approach [35] achieves this verification by checking the three inter-star angles (only two of which are independent) between the three stars forming the triangle match hypothesis and a fourth star—these four stars from the so-called *pyramid* from which this algorithm derives its name. The Astrometry.net approach [3] achieves this verification by using the match hypothesis to compute the attitude, reprojecting catalog stars onto the image, and making additional star correspondences until a Bayes factor metric exceeds a conservative threshold.

V. ATTITUDE DETERMINATION

The star identification procedure described in Sec. IV allows for the matching of stars observed in an image with known stars in a catalog. These corresponding lists of observed and catalog stars may be used to determine the attitude of the camera when the image was captured. The simplest case is a calibrated camera and this situation is discussed first. We then consider the case of a fully uncalibrated camera, followed by the special case when only the ratio if focal length to pixel pitch ($d_x = f/\mu_x$) is unknown.

A. CALIBRATED CAMERA

When the camera is calibrated, the star locations in the image may be converted directly to unit vector directions as discussed in Section III-A. In this case, the attitude determination problem is to find the proper orthogonal matrix $\boldsymbol{T}$ that solves Wahba's problem [14],

$$\min_{\boldsymbol{T}\in\mathrm{SO}(3)} L(\boldsymbol{T}) = \frac{1}{2}\sum_{i=1}^{n} w_i \|\boldsymbol{a}_i - \boldsymbol{T}\boldsymbol{e}_i\|^2 \tag{33}$$

where $w_i > 0$ is the weight for the ith measurement. Though, there are a variety of solutions to this problem [13], this work briefly reviews the solution with the singular value decomposition (SVD). That the SVD provided a solution to Wahba's problem was immediately apparent [61] and was quite mature by the late 1980s [12].

The most compact derivation, and the one we prefer, is to rewrite Wahba's problem as an orthogonal Procrustes problem [62]—which has a known solution with the SVD. If the ICRF star directions and measured star directions are arranged as columns in the matrices $\boldsymbol{E}$ and $\boldsymbol{A}$, respectively,

$$\boldsymbol{E} = \begin{bmatrix}\boldsymbol{e}_1 & \dots & \boldsymbol{e}_n\end{bmatrix} \quad \text{and} \quad \boldsymbol{A} = \begin{bmatrix}\boldsymbol{a}_1 & \dots & \boldsymbol{a}_n\end{bmatrix} \tag{34}$$

then, from Eq. (2), it follows that

$$\boldsymbol{A} = \boldsymbol{T}\boldsymbol{E} \tag{35}$$

Moreover, defining, a diagonal weight matrix $\boldsymbol{W} = \boldsymbol{W}^{\frac{1}{2}}\boldsymbol{W}^{\frac{1}{2}}$ as

$$\boldsymbol{W} = \mathrm{diag}(w_1, \dots, w_n) \tag{36}$$

$$\boldsymbol{W}^{\frac{1}{2}} = \mathrm{diag}(\sqrt{w_1}, \dots, \sqrt{w_n}) \tag{37}$$

we may rewrite the Wahba problem from Eq. (33) as

$$\min_{\boldsymbol{T}\in\mathrm{SO}(3)} L(\boldsymbol{T}) = \frac{1}{2}\|\boldsymbol{A}\boldsymbol{W}^{\frac{1}{2}} - \boldsymbol{T}\boldsymbol{E}\boldsymbol{W}^{\frac{1}{2}}\|_F^2$$

where $\|\cdot\|_F$ is the matrix Frobenius norm. The matrix optimization problem on the right-hand side is an *orthogonal Procrustes problem*—which arises in many applied mathematics problems [62]—and was famously solved in the mid-1960s [63].[1] The solution to the orthogonal Procrustes problem is to compute the matrix $\boldsymbol{B}$

$$\boldsymbol{B} = \left(\boldsymbol{A}\boldsymbol{W}^{\frac{1}{2}}\right)\left(\boldsymbol{E}\boldsymbol{W}^{\frac{1}{2}}\right)^T = \boldsymbol{A}\boldsymbol{W}\boldsymbol{E}^T = \sum_{i=1}^{n} w_i \boldsymbol{a}_i \boldsymbol{e}_i^T \tag{38}$$

and then compute the SVD $\boldsymbol{B} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^T$. From here, the optimal solution for the attitude transformation matrix $\boldsymbol{T}$ is

$$\hat{\boldsymbol{T}} = \boldsymbol{U}\begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & \det(\boldsymbol{U})\det(\boldsymbol{V})\end{bmatrix}\boldsymbol{V}^T \tag{39}$$

The term $\det(\boldsymbol{U})\det(\boldsymbol{V})$ is necessary to ensure that the attitude transformation matrix is proper orthogonal [i.e., that $\det(\boldsymbol{T}) = +1$] and describes a right-handed coordinate system [12].

This result is well-known and is repeated here to (1) better highlight the orthogonal Procrustes problem interpretation of the SVD-based solution and (2) to provide a point of contrast/comparison for attitude determination with an uncalibrated camera (which also uses the SVD).

[1] There are earlier solutions to this problem, such as Green's solution developed in 1952 [64], but Schönemann's SVD-based approach from [63] is elegant, robust, and the most widely used today.

B. UNCALIBRATED CAMERA (THE GENERIC CASE)

The appearance of an asterism in an image is related to the catalog (or the asterism's appearance in another image) through a homography. This is a direct result of Eq. (10).

Therefore, given the homogrpahy from Eq. (10), let the contributing matrix/vector entries be given by

$$\begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} \propto \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} \tag{40}$$

Further, remove the scale ambiguity (different for each star) by introducing the unknown scalar s_i such that Eq. (10) becomes an equality relation

$$s_i \bar{\boldsymbol{u}}_i = \boldsymbol{H} \boldsymbol{e}_i \tag{41}$$

The scalar s_i is different for each star observation and may be computed as

$$s_i = \boldsymbol{k}^T \boldsymbol{H} \boldsymbol{e}_i \tag{42}$$

where $\boldsymbol{k}^T = [0\ 0\ 1]$. Ideally, the loss function we would like to solve is the maximum likelihood estimate (MLE) given by

$$\min_{\boldsymbol{H} \in \mathrm{PGL}(3)} L(\boldsymbol{H}) = \frac{1}{2} \sum_{i=1}^{n} \delta \boldsymbol{u}_i^T \boldsymbol{R}_i^{-1} \delta \boldsymbol{u}_i \tag{43}$$

where

$$\delta \boldsymbol{u}_i = \tilde{\boldsymbol{u}}_i - \frac{\boldsymbol{S} \boldsymbol{H} \boldsymbol{e}_i}{\boldsymbol{k}^T \boldsymbol{H} \boldsymbol{e}_i}$$

and where $\tilde{\boldsymbol{u}}_i^T = [\tilde{u}_i, \tilde{v}_i]$ is the measured image coordinate of the ith star, $\boldsymbol{S} = [\boldsymbol{I}_2\ \ \boldsymbol{0}_{2\times 1}]$, and $\boldsymbol{R}_i$ is the 2×2 covariance matrix describing the star centroiding error in the image. Let the elements of the measurement covariance $\boldsymbol{R}_i$ be given by

$$\boldsymbol{R}_i = \begin{bmatrix} \sigma_{u_i}^2 & \sigma_{uv_i} \\ \sigma_{uv_i} & \sigma_{v_i}^2 \end{bmatrix} \tag{44}$$

This loss function is desirable since it minimizes the residuals of the reprojection of the stars in the image, which is where the measurements are made. The difficulty is that this is a nonlinear optimization problem that requires iteration to solve.

That s_i [see Eq. (42)] is different for each star and also depends on $\boldsymbol{H}$ is problematic. It may be eliminated from the problem using the so-called Direct Linear Transform (DLT) [25]. For this problem, the DLT is accomplished by taking the cross-product of both sides of Eq. (41) with the star image coordinates $\bar{\boldsymbol{u}}_i$,

$$s_i [\bar{\boldsymbol{u}}_i \times] \bar{\boldsymbol{u}}_i = [\bar{\boldsymbol{u}}_i \times] \boldsymbol{H} \boldsymbol{e}_i = \boldsymbol{0}_{3\times 1} \tag{45}$$

where $[\,\cdot\,\times]$ is the skew symmetric cross-product matrix such that $[\boldsymbol{a}\times]\boldsymbol{b} = \boldsymbol{a} \times \boldsymbol{b}$. To isolate the elements of $\boldsymbol{H}$ in an easily solvable form, rewrite the non-trivial (middle) term with a Kronecker product,

$$\begin{aligned} [\bar{\boldsymbol{u}}_i \times] \boldsymbol{H} \boldsymbol{e}_i &= \left(\boldsymbol{e}_i^T \otimes [\bar{\boldsymbol{u}}_i \times] \right) \mathrm{vec}(\boldsymbol{H}) \\ &= \left([\bar{\boldsymbol{u}}_i \times] \otimes \boldsymbol{e}_i^T \right) \mathrm{vec}(\boldsymbol{H}^T) \end{aligned} \tag{46}$$

where $\otimes$ is the Kronocker product operator. The 3×9 matrix acting on $\mathrm{vec}(\boldsymbol{H}^T)$ may be expanded as

$$[\bar{\boldsymbol{u}}_i \times] \otimes \boldsymbol{e}_i^T = \begin{bmatrix} 0 & -\boldsymbol{e}_i^T & v_i \boldsymbol{e}_i^T \\ \boldsymbol{e}_i^T & 0 & -u_i \boldsymbol{e}_i^T \\ -v_i \boldsymbol{e}_i^T & u_i \boldsymbol{e}_i^T & 0 \end{bmatrix} \tag{47}$$

Noting that $[\bar{\boldsymbol{u}}_i \times]$ is rank 2, only two rows of $[\bar{\boldsymbol{u}}_i \times] \otimes \boldsymbol{e}_i^T$ are independent. Thus, using only the first two rows

$$\begin{bmatrix} \boldsymbol{0}_{1\times 3} & -\boldsymbol{e}_i^T & v_i \boldsymbol{e}_i^T \\ \boldsymbol{e}_i^T & \boldsymbol{0}_{1\times 3} & -u_i \boldsymbol{e}_i^T \end{bmatrix} \boldsymbol{h} = \boldsymbol{0}_{2\times 1} \tag{48}$$

where $\boldsymbol{h}$ is given by

$$\begin{aligned} \boldsymbol{h} &= \mathrm{vec}(\boldsymbol{H}^T) \\ &= \begin{bmatrix} h_{11} & h_{12} & h_{13} & h_{21} & h_{22} & h_{23} & h_{31} & h_{32} & h_{33} \end{bmatrix}^T \end{aligned} \tag{49}$$

Observing that the right-hand side is zero, the second row may be multiplied by -1 and the rows swapped to produce the equivalent (and more convenient) expression

$$\begin{bmatrix} -\boldsymbol{e}_i^T & \boldsymbol{0}_{1\times 3} & u_i \boldsymbol{e}_i^T \\ \boldsymbol{0}_{1\times 3} & -\boldsymbol{e}_i^T & v_i \boldsymbol{e}_i^T \end{bmatrix} \boldsymbol{h} = \boldsymbol{D}_i \boldsymbol{h} = \boldsymbol{0}_{2\times 1} \tag{50}$$

and where $\boldsymbol{D}_i$ is a 2×9 of rank 2. When measurements of star coordinates in the image are noisy, $[\tilde{u}_i, \tilde{v}_i]$, this constraint is not exactly satisfied and we obtain

$$\begin{bmatrix} -\boldsymbol{e}_i^T & \boldsymbol{0}_{1\times 3} & \tilde{u}_i \boldsymbol{e}_i^T \\ \boldsymbol{0}_{1\times 3} & -\boldsymbol{e}_i^T & \tilde{v}_i \boldsymbol{e}_i^T \end{bmatrix} \boldsymbol{h} = \tilde{\boldsymbol{D}}_i \boldsymbol{h} = \boldsymbol{\epsilon} \tag{51}$$

where $\tilde{\boldsymbol{D}}_i$ is the noisy version of $\boldsymbol{D}_i$.

1) THE DIRECT LINEAR TRANSFORM (DLT) SOLUTION FOR H

This motivates an alternate loss function to solve for $\boldsymbol{H}$ based on minimization of the DLT residuals,

$$\min_{\boldsymbol{H} \in \mathrm{PGL}(3)} L(\boldsymbol{H}) = \frac{1}{2} \sum_{i=1}^{n} \boldsymbol{\epsilon}^T \boldsymbol{\epsilon} \tag{52}$$

Substitution from Eq. (51) leads to

$$\min_{\boldsymbol{H} \in \mathrm{PGL}(3)} L(\boldsymbol{H}) = \frac{1}{2} \sum_{i=1}^{n} \boldsymbol{h}^T \tilde{\boldsymbol{D}}_i^T \tilde{\boldsymbol{D}}_i \boldsymbol{h} \tag{53}$$

For an asterism of n stars, the optimal estimate of the homography using the DLT loss function may be found by taking the first differential of Eq. (53) and setting the result to zero

$$\sum_{i=1}^{n} \boldsymbol{h}^T \tilde{\boldsymbol{D}}_i^T \tilde{\boldsymbol{D}}_i = \boldsymbol{0}_{1\times 9} \tag{54}$$

which is equivalent to

$$\left(\sum_{i=1}^{n} \tilde{\boldsymbol{D}}_i^T \tilde{\boldsymbol{D}}_i \right) \boldsymbol{h} = \boldsymbol{0}_{9\times 1} \tag{55}$$

Therefore, it is evident that $\boldsymbol{h}$ lies in the null space of $\sum \tilde{\boldsymbol{D}}_i^T \tilde{\boldsymbol{D}}_i$. Unfortunately, the numerical conditioning of $\sum \tilde{\boldsymbol{D}}_i^T \tilde{\boldsymbol{D}}_i$ is sometimes poor. Therefore, define the matrix $\tilde{\boldsymbol{D}}$

$$\tilde{\boldsymbol{D}} = \begin{bmatrix} \tilde{\boldsymbol{D}}_1 \\ \vdots \\ \tilde{\boldsymbol{D}}_n \end{bmatrix} \tag{56}$$

such that

$$\sum_{i=1}^{n} \tilde{\boldsymbol{D}}_i^T \tilde{\boldsymbol{D}}_i = \tilde{\boldsymbol{D}}^T \tilde{\boldsymbol{D}} \tag{57}$$

Since the null space of $\tilde{\boldsymbol{D}}^T \tilde{\boldsymbol{D}}$ is the same as the null space of $\tilde{\boldsymbol{D}}$, superior numerical conditioning may be achieved by solving the linear system

$$\tilde{\boldsymbol{D}}\boldsymbol{h} = \boldsymbol{0}_{2n\times 1} \tag{58}$$

which produces the same result for $\boldsymbol{h}$ as Eq. (55). In a noise-free scenario with $n \geq 4$, the $2n \times 9$ matrix $\tilde{\boldsymbol{D}} = \boldsymbol{D}$ is rank 8 with a 1D null space defined by $\boldsymbol{h}$. In the presence of noise and with $n \geq 5$, the matrix $\tilde{\boldsymbol{D}}$ will generally be rank 9 and we may find $\boldsymbol{h}$ through singular value decomposition, $\boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^T = \tilde{\boldsymbol{D}}$, with $\boldsymbol{h}$ being the column of $\boldsymbol{V}$ corresponding to the smallest singluar value. Therefore, the 9×1 vector $\boldsymbol{h}$ (which has arbitrary scale) may be found directly from image-to-catalog star correspondences and fully defines the 3×3 homography matrix $\boldsymbol{H}$ (which also has arbitrary scale).

Despite being better than $\sum \tilde{\boldsymbol{D}}_i^T \tilde{\boldsymbol{D}}_i$, the numerical conditioning of $\tilde{\boldsymbol{D}}$ can be still quite poor for a high-resolution focal plane array. This is a common problem with DLT solutions for the homography matrix. Therefore, making use of Hartley's normalization scheme from Ref. [65], consider the transformation

$$\bar{\boldsymbol{u}}_i' = \boldsymbol{M}\bar{\boldsymbol{u}}_i \tag{59}$$

where $\boldsymbol{M}$ is a chosen such that the centroid of $\{\bar{\boldsymbol{u}}_i'\}_{i=1}^n$ is the origin and the mean distance is $\sqrt{2}$. The new homography matrix is $\boldsymbol{H}' = \boldsymbol{M}\boldsymbol{H}$ and

$$s_i\bar{\boldsymbol{u}}_i' = \boldsymbol{M}\boldsymbol{H}\boldsymbol{e}_i = \boldsymbol{H}'\boldsymbol{e}_i \tag{60}$$

Therefore, better numerical performance is generally achieved by solving the problem

$$\tilde{\boldsymbol{D}}'\boldsymbol{h}' = \boldsymbol{0}_{2n\times 1} \tag{61}$$

where the solution $\boldsymbol{h}' = \text{vec}(\boldsymbol{H}'^T)$ may be rearranged to find $\boldsymbol{H}'$. Then, the homography is given by

$$\boldsymbol{H} = \boldsymbol{M}^{-1}\boldsymbol{H}' \tag{62}$$

This approach generally provides superior performance as compared to solving the problem without normalization.

2) THE TOTAL LEAST SQUARES (TLS) SOLUTION FOR $\boldsymbol{H}$

Equation (51) represents a total least squares (TLS) model [66] and it is not self-evident that the DLT solution will necessarily provide the statistically optimal estimate of the homography matrix. Therefore, we now develop the statistically rigorous TLS solution.

To proceed it is useful to rewrite the homography constraint from Eq. (50) as

$$\boldsymbol{D}_i\boldsymbol{h} = (\boldsymbol{h}^T \otimes \boldsymbol{I}_2)\boldsymbol{d}_i = \boldsymbol{0}_{2\times 1} \tag{63}$$

where $\boldsymbol{h} = \text{vec}(\boldsymbol{H}^T)$ from Eq. (49) and where $\boldsymbol{d}_i$ is the 18×1 vector

$$\boldsymbol{d}_i = \begin{bmatrix} \boldsymbol{e}_i \otimes \begin{bmatrix} -1 \\ 0 \end{bmatrix} \\ \boldsymbol{e}_i \otimes \begin{bmatrix} 0 \\ -1 \end{bmatrix} \\ \boldsymbol{e}_i \otimes \begin{bmatrix} v_i \\ u_i \end{bmatrix} \end{bmatrix} \tag{64}$$

Likewise, also define $\tilde{\boldsymbol{d}}_i$ and $\hat{\boldsymbol{d}}_i$ as:

$$\tilde{\boldsymbol{d}}_i = \begin{bmatrix} \boldsymbol{e}_i \otimes \begin{bmatrix} -1 \\ 0 \end{bmatrix} \\ \boldsymbol{e}_i \otimes \begin{bmatrix} 0 \\ -1 \end{bmatrix} \\ \boldsymbol{e}_i \otimes \begin{bmatrix} \tilde{v}_i \\ \tilde{u}_i \end{bmatrix} \end{bmatrix}, \quad \hat{\boldsymbol{d}}_i = \begin{bmatrix} \boldsymbol{e}_i \otimes \begin{bmatrix} -1 \\ 0 \end{bmatrix} \\ \boldsymbol{e}_i \otimes \begin{bmatrix} 0 \\ -1 \end{bmatrix} \\ \boldsymbol{e}_i \otimes \begin{bmatrix} \hat{v}_i \\ \hat{u}_i \end{bmatrix} \end{bmatrix} \tag{65}$$

The covariance of $\tilde{\boldsymbol{d}}_i$ is given by

$$\boldsymbol{R}_{\boldsymbol{d}_i} = \begin{bmatrix} \boldsymbol{0}_{12\times 12} & \boldsymbol{0}_{12\times 6} \\ \boldsymbol{0}_{6\times 12} & \boldsymbol{e}_i\boldsymbol{e}_i^T \otimes \boldsymbol{R}_i \end{bmatrix} \tag{66}$$

where $\boldsymbol{R}_i$ is from Eq (44). Note that $\boldsymbol{R}_{\boldsymbol{d}_i}$ has rank 2 because $\text{rank}(\boldsymbol{e}_i\boldsymbol{e}_i^T \otimes \boldsymbol{R}_i) = \text{rank}(\boldsymbol{e}_i\boldsymbol{e}_i^T) \times \text{rank}(\boldsymbol{R}_i)$. Thus, the TLS loss function is given by

$$L(\hat{\boldsymbol{d}}_i) = \frac{1}{2}\sum_{i=1}^{n}(\tilde{\boldsymbol{d}}_i - \hat{\boldsymbol{d}}_i)^T\boldsymbol{R}_{\boldsymbol{d}_i}^{-1}(\tilde{\boldsymbol{d}}_i - \hat{\boldsymbol{d}}_i), \quad \text{subject to } (\hat{\boldsymbol{h}}^T \otimes \boldsymbol{I}_2)\hat{\boldsymbol{d}}_i = \boldsymbol{0}_{2\times 1}, \quad i = 1, 2, \ldots, n \tag{67}$$

The constrained loss function is rewritten into an unconstrained one by determining a solution for $\hat{\boldsymbol{d}}_i$ and substituting its result back into Eq. (67). To accomplish this task, the constraint is appended to the loss function using Lagrange multipliers,

$$L(\hat{\boldsymbol{d}}_i) = \sum_{i=1}^{n}\boldsymbol{\lambda}_i^T(\hat{\boldsymbol{h}}^T \otimes \boldsymbol{I}_2)\hat{\boldsymbol{d}}_i + \frac{1}{2}(\tilde{\boldsymbol{d}}_i - \hat{\boldsymbol{d}}_i)^T\boldsymbol{R}_{\boldsymbol{d}_i}^{-1}(\tilde{\boldsymbol{d}}_i - \hat{\boldsymbol{d}}_i) \tag{68}$$

where each $\boldsymbol{\lambda}_i$ is a 2×1 Lagrange multiplier. Taking the partial of Eq. (68) with respect to each $\hat{\boldsymbol{d}}_i$ leads to the following n necessary conditions:

$$\boldsymbol{R}_{d_i}^{-1}\hat{\boldsymbol{d}}_i - \boldsymbol{R}_{d_i}^{-1}\tilde{\boldsymbol{d}}_i + (\hat{\boldsymbol{h}}\otimes \boldsymbol{I}_2)\boldsymbol{\lambda}_i = \boldsymbol{0}_{9\times 1} \tag{69}$$

Left multiplying Eq. (69) by $(\hat{\boldsymbol{h}}^T \otimes \boldsymbol{I}_2)\boldsymbol{R}_{d_i}$, using the constraint $(\hat{\boldsymbol{h}}^T \otimes \boldsymbol{I}_2)\hat{\boldsymbol{d}}_j = \boldsymbol{0}_{2\times 1}$, and solving for $\boldsymbol{\lambda}_i$ gives

$$\boldsymbol{\lambda}_i = \boldsymbol{Q}_i^{-1}(\hat{\boldsymbol{h}}^T \otimes \boldsymbol{I}_2)\tilde{\boldsymbol{d}}_i \tag{70}$$

where

$$\boldsymbol{Q}_i \equiv (\hat{\boldsymbol{h}}^T \otimes \boldsymbol{I}_2)\boldsymbol{R}_{d_i}(\hat{\boldsymbol{h}}\otimes \boldsymbol{I}_2) \tag{71}$$

Substituting Eq. (70) into Eq. (69) leads to

$$\hat{\boldsymbol{d}}_i = \left[\boldsymbol{I}_{18} - \boldsymbol{R}_{d_i}(\hat{\boldsymbol{h}}\otimes \boldsymbol{I}_2)\boldsymbol{Q}_i^{-1}(\hat{\boldsymbol{h}}^T \otimes \boldsymbol{I}_2)\right]\tilde{\boldsymbol{d}}_i \tag{72}$$

Further substituting Eq. (72) into Eq. (67) now gives the unconstrained loss function

$$L(\hat{\boldsymbol{h}}) = \frac{1}{2}\sum_{i=1}^{n}\tilde{\boldsymbol{d}}_i^T(\hat{\boldsymbol{h}}\otimes \boldsymbol{I}_2)\boldsymbol{Q}_i^{-1}(\hat{\boldsymbol{h}}^T \otimes \boldsymbol{I}_2)\tilde{\boldsymbol{d}}_i \tag{73}$$

which is now only a function of the unknown $\hat{\boldsymbol{h}}$. The reader is reminded that $\boldsymbol{h}$ has an unknown scale and, as expected we observe that $L(\hat{\boldsymbol{h}}) = L(\alpha\hat{\boldsymbol{h}})$ for all $\alpha \neq 0$. The most common solution is to choose $\|\hat{\boldsymbol{h}}\| = 1$, but other choices are equally valid. Note that even though $\boldsymbol{R}_{d_i}$ is a singular matrix, the 2×2 matrix $\boldsymbol{Q}_i$ is generally non-singular. Oftentimes, the solution using Eq. (58) or the approximate solution discussed below may be adequate, making the formal minimization of Eq. (73) unnecessary. When an exact solution to the loss function in Eq. (73) is required, one may use the iterative solution provided in Appendix B of Ref. [67].

3) AN APPROXIMATE NON-ITERATIVE TLS SOLUTION FOR H

An approximate non-iterative solution for $\hat{\boldsymbol{h}}$ may be found by assuming that the covariance for each row of the matrix $\tilde{\boldsymbol{D}}$ is the same. This is nearly true in many practical cases and the details are now shown.

For many cameras and image processing techniques, it is reasonable to assume that star centroiding errors are uncorrelated in the u and v directions: $\sigma_{uv_i} = 0$. Moreover, under the assumption that centroiding errors in the u and v directions are the same, we find that $\sigma^2 \approx \sigma_{u_i}^2 \approx \sigma_{v_i}^2$. These assumptions lead to the approximation of the star centroid covariance from Eq. (44) as $\boldsymbol{R}_i \approx \sigma^2\boldsymbol{I}_2$.

For cameras with a narrow FOV, all the star observations are clustered in the same approximate direction. In this case, we find that

$$\bar{\boldsymbol{R}}_e \approx \sigma_{u_i}^2\boldsymbol{e}_i\boldsymbol{e}_i^T \approx \sigma_{v_i}^2\boldsymbol{e}_i\boldsymbol{e}_i^T, \quad i = 1,\ldots,n \tag{74}$$

where $\bar{\boldsymbol{R}}_e$ is

$$\bar{\boldsymbol{R}}_e \equiv \frac{\sigma^2}{n}\sum_{i=1}^{n}\boldsymbol{e}_i\boldsymbol{e}_i^T \tag{75}$$

Therefore, to a good approximation, the covariance for every row of the matrix $\tilde{\boldsymbol{D}}$ is about the same and is given by the constant matrix

$$\bar{\boldsymbol{R}}_D \equiv \begin{bmatrix}\boldsymbol{0}_{6\times 6} & \boldsymbol{0}_{6\times 3}\\ \boldsymbol{0}_{3\times 6} & \bar{\boldsymbol{R}}_e\end{bmatrix} \tag{76}$$

Now, partition the matrix $\tilde{\boldsymbol{D}}$ into two columns,

$$\tilde{\boldsymbol{D}} = \begin{bmatrix}\tilde{\boldsymbol{D}}_1\\ \tilde{\boldsymbol{D}}_2\\ \vdots\\ \tilde{\boldsymbol{D}}_n\end{bmatrix} \equiv \begin{bmatrix}\boldsymbol{\Gamma}_1 & \boldsymbol{\Gamma}_2\end{bmatrix} \tag{77}$$

where $\boldsymbol{\Gamma}_1$ is a matrix made up of the first six columns of $\tilde{\boldsymbol{D}}$, and $\boldsymbol{\Gamma}_2$ is a matrix made up of the last three columns of $\tilde{\boldsymbol{D}}$. Assuming a high-quality star catalog (as compared to star tracker measurement noise), we assume the elements of $\boldsymbol{\Gamma}_1$ are known perfectly. It is also assumed that each row of $\boldsymbol{\Gamma}_2$ has covariance given by $\bar{\boldsymbol{R}}_e$. This is consistent with the assumption that each row of $\tilde{\boldsymbol{D}}$ has a covariance given by $\bar{\boldsymbol{R}}_D$ from Eq. (76). This now corresponds to the element-wise uncorrelated and stationary TLS problem [66]. The property of having some elements of $\tilde{\boldsymbol{D}}$ known perfectly while others containing errors is known as the generalized TLS (GTLS) problem, and a computationally efficient and numerically reliable non-iterative algorithm to solve the GTLS problem is shown in Ref. [68]. This algorithm is preferred over others because $\bar{\boldsymbol{R}}_e$ may be nearly singular, especially for small field-of-view cameras.

4) FINDING K AND T FROM THE HOMOGRAPHY H

Given an estimate of $\boldsymbol{H}$ (e.g., from the DLT or GTLS solution), it is necessary to split $\boldsymbol{H}$ into $\boldsymbol{K}$ and $\boldsymbol{T}$ if the objective is attitude determination. It was suggested in Ref. [69] that such a task requires three starfield images, but this is not true. The fully generic calibration may be performed with only a single image. To see this, we may employ a modification of Zhang's classic self-calibration technique with a planar pattern [70]. Since the present problem is one of viewing star directions (not a plane) and of attitude estimation (and not pose estimation), the framework can be modified with little effort to obtain simultaneous self-calibration and attitude estimation with a single image. The details are now discussed.

To begin, partition the attitude transformation matrix and homography matrix by columns as

$$\boldsymbol{T} = \begin{bmatrix}\boldsymbol{t}_1 & \boldsymbol{t}_2 & \boldsymbol{t}_3\end{bmatrix} \quad \text{and} \quad \boldsymbol{H} = \begin{bmatrix}\boldsymbol{h}_1 & \boldsymbol{h}_2 & \boldsymbol{h}_3\end{bmatrix} \tag{78}$$

such that we may rewrite Eq. (9) as

$$\boldsymbol{H} = \begin{bmatrix}\boldsymbol{h}_1 & \boldsymbol{h}_2 & \boldsymbol{h}_3\end{bmatrix} = \omega\boldsymbol{K}\begin{bmatrix}\boldsymbol{t}_1 & \boldsymbol{t}_2 & \boldsymbol{t}_3\end{bmatrix} \tag{79}$$

where $\omega \neq 0$ describes the arbitrary scale of $\boldsymbol{H}$. Therefore, for each column,

$$\boldsymbol{h}_i = \omega\boldsymbol{K}\boldsymbol{t}_i, \quad i = 1,2,3 \tag{80}$$

or, since $\boldsymbol{K}$ is always full rank,

$$\boldsymbol{t}_i = \frac{1}{\omega}\boldsymbol{K}^{-1}\boldsymbol{h}_i, \quad i = 1, 2, 3 \tag{81}$$

The attitude transformation is an orthonormal matrix, leading to six constraints on the vectors $\boldsymbol{t}_1, \boldsymbol{t}_2, \boldsymbol{t}_3$. The first three constraints ensure that the columns are orthogonal: $\boldsymbol{t}_i^T\boldsymbol{t}_j = 0$ for $i \neq j$. Therefore,

$$\boldsymbol{t}_i^T\boldsymbol{t}_j = \boldsymbol{h}_i^T\boldsymbol{K}^{-T}\boldsymbol{K}^{-1}\boldsymbol{h}_j = 0 \tag{82}$$

To compact notation, define $\boldsymbol{B} = \boldsymbol{K}^{-T}\boldsymbol{K}^{-1}$ as the 3×3 matrix

$$\boldsymbol{B} = \boldsymbol{K}^{-T}\boldsymbol{K}^{-1} = \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{12} & b_{22} & b_{23} \\ b_{13} & b_{23} & b_{33} \end{bmatrix} \tag{83}$$

Consequently, the constraint of Eq. (82) may be rewritten as

$$\boldsymbol{t}_i^T\boldsymbol{t}_j = \boldsymbol{h}_i^T\boldsymbol{B}\boldsymbol{h}_j = 0 \rightarrow \left(\boldsymbol{h}_j \otimes \boldsymbol{h}_i\right)^T \text{vec}(\boldsymbol{B}) = 0 \tag{84}$$

where $\otimes$ is the Kronocker product. Note that $\boldsymbol{B}$ is a 3×3 symmetric matrix, such that vec($\boldsymbol{B}$) has only six independent elements. Therefore, define the 6×1 vector $\boldsymbol{b} = [b_{11}, b_{12}, b_{22}, b_{13}, b_{23}, b_{33}]^T$, such that

$$\left(\boldsymbol{h}_j \otimes \boldsymbol{h}_i\right)^T \text{vec}(\boldsymbol{B}) = 0 \quad \rightarrow \quad \boldsymbol{v}_{ij}^T\boldsymbol{b} = 0 \tag{85}$$

where

$$\boldsymbol{v}_{ij} = \begin{bmatrix} h_{1j}h_{1i} \\ h_{1j}h_{2i} + h_{2j}h_{1i} \\ h_{2j}h_{2i} \\ h_{1j}h_{3i} + h_{3j}h_{1i} \\ h_{2j}h_{3i} + h_{3j}h_{2i} \\ h_{3j}h_{3i} \end{bmatrix} \tag{86}$$

This leads to three expressions for the unique combinations of $\boldsymbol{h}_1, \boldsymbol{h}_2, \boldsymbol{h}_3$,

$$\begin{aligned} \boldsymbol{v}_{12}^T\boldsymbol{b} &= 0 \\ \boldsymbol{v}_{13}^T\boldsymbol{b} &= 0 \\ \boldsymbol{v}_{23}^T\boldsymbol{b} &= 0 \end{aligned} \tag{87}$$

Likewise, there are three more constraints to ensure that the columns of $\boldsymbol{T}$ are unit vectors: $\boldsymbol{t}_i^T\boldsymbol{t}_i = 1$. This yields

$$\boldsymbol{t}_i^T\boldsymbol{t}_i = \frac{1}{\omega^2}\boldsymbol{h}_i^T\boldsymbol{K}^{-T}\boldsymbol{K}^{-1}\boldsymbol{h}_i = 1 \tag{88}$$

$$\boldsymbol{h}_i^T\boldsymbol{B}\boldsymbol{h}_i = \omega^2 \tag{89}$$

To remove the dependence on ω, column inner products may be compared to one another

$$\boldsymbol{h}_i^T\boldsymbol{B}\boldsymbol{h}_i = \omega^2 = \boldsymbol{h}_j^T\boldsymbol{B}\boldsymbol{h}_j \tag{90}$$

or, equivalently,

$$(\boldsymbol{h}_i \otimes \boldsymbol{h}_i)^T \text{vec}(\boldsymbol{B}) = \left(\boldsymbol{h}_j \otimes \boldsymbol{h}_j\right)^T \text{vec}(\boldsymbol{B}) \tag{91}$$

$$\left[(\boldsymbol{h}_i \otimes \boldsymbol{h}_i) - \left(\boldsymbol{h}_j \otimes \boldsymbol{h}_j\right)\right]^T \text{vec}(\boldsymbol{B}) = 0 \tag{92}$$

$$\left(\boldsymbol{v}_{ii} - \boldsymbol{v}_{jj}\right)^T \boldsymbol{b} = 0 \tag{93}$$

This produces three more expressions for the unique combinations of $\boldsymbol{h}_1, \boldsymbol{h}_2, \boldsymbol{h}_3$,

$$\begin{aligned} (\boldsymbol{v}_{11} - \boldsymbol{v}_{22})^T\boldsymbol{b} &= 0 \\ (\boldsymbol{v}_{11} - \boldsymbol{v}_{33})^T\boldsymbol{b} &= 0 \\ (\boldsymbol{v}_{22} - \boldsymbol{v}_{33})^T\boldsymbol{b} &= 0 \end{aligned} \tag{94}$$

The six constraints may be stacked into a linear system

$$\begin{bmatrix} \boldsymbol{v}_{12}^T \\ \boldsymbol{v}_{13}^T \\ \boldsymbol{v}_{23}^T \\ (\boldsymbol{v}_{11} - \boldsymbol{v}_{22})^T \\ (\boldsymbol{v}_{11} - \boldsymbol{v}_{33})^T \\ (\boldsymbol{v}_{22} - \boldsymbol{v}_{33})^T \end{bmatrix} \boldsymbol{b} \equiv \boldsymbol{V}_h\boldsymbol{b} = \boldsymbol{0}_{6\times 1} \tag{95}$$

which may be solved for $\boldsymbol{b}$ (to an unknown scale) in either the least squares sense or the TLS sense. In most practical cases, the optimal value of $\boldsymbol{b}$ may be found by finding (or approximating) the null space of $\boldsymbol{V}_h$ via a SVD. The procedure is identical to the SVD-based solution for Eq. (58).

To formally handle the measurement statistics, however, requires minimization of the TLS loss function for the estimate of $\boldsymbol{b}$, denoted by $\hat{\boldsymbol{b}}$. Development follows along the lines leading to Eq. (73), with the present problem taking the form

$$L(\hat{\boldsymbol{b}}) = \frac{1}{2}\text{vec}^T(\boldsymbol{V}_h^T)(\boldsymbol{I}_6 \otimes \hat{\boldsymbol{b}})\boldsymbol{Q}_h^{-1}(\boldsymbol{I}_6 \otimes \hat{\boldsymbol{b}}^T)\text{vec}(\boldsymbol{V}_h^T) \tag{96}$$

where

$$\boldsymbol{Q}_h \equiv (\boldsymbol{I}_6 \otimes \hat{\boldsymbol{b}}^T)\boldsymbol{P}_v(\boldsymbol{I}_6 \otimes \hat{\boldsymbol{b}}) \tag{97}$$

and $\boldsymbol{P}_v$ is the error-covariance of vec($\boldsymbol{V}_h^T$), which is shown later [see Eq. (117)]. Equation (96) may also have several local minimums, which can be mitigated by applying the constraint $\|\hat{\boldsymbol{b}}\| = 1$. It is rare that we find the TLS solution to be meaningfully different than the simpler SVD-based solution.

The five independent values of the estimate $\hat{\boldsymbol{b}}$ may be used to find the five standard camera calibration parameters $d_x, d_y, \alpha, u_p, v_p$. This may be done by rewriting $\hat{\boldsymbol{b}}$ as $\hat{\boldsymbol{B}}$ and returning to the original equation,

$$\boldsymbol{B} = \eta\boldsymbol{K}^{-T}\boldsymbol{K}^{-1} \tag{98}$$

where the new unknown η represents the ambiguous scale. Recovering the five camera calibration parameters that comprise $\boldsymbol{K}$ when given only an estimate of $\boldsymbol{B}$ was first accomplished by Zhang [70] and these results may be used here without modification. Therefore, after some tedious algebra (left as an exercise to the reader), we find that

$$v_p = \frac{b_{12}b_{13} - b_{11}b_{23}}{b_{11}b_{22} - b_{12}^2} \tag{99a}$$

$$\eta = b_{33} - \frac{1}{b_{11}}\left[b_{13}^2 + v_p\left(b_{12}b_{13} - b_{11}b_{23}\right)\right] \tag{99b}$$

$$d_x = \sqrt{\frac{\eta}{b_{11}}} \tag{99c}$$

$$d_y = \sqrt{\frac{\eta b_{11}}{b_{11}b_{22} - b_{12}^2}} \tag{99d}$$

$$\alpha = -\frac{1}{\eta} b_{12} d_x^2 d_y \tag{99e}$$

$$u_p = \frac{\alpha v_p}{d_y} - \frac{b_{13} d_x^2}{\eta} \tag{99f}$$

With the five elements of $\boldsymbol{K}$ known,

$$\omega \boldsymbol{T} = \boldsymbol{K}^{-1} \boldsymbol{H} \tag{100}$$

which is rearranged to find

$$\omega \boldsymbol{K}^T = \boldsymbol{T} \boldsymbol{H}^T \tag{101}$$

where ω and $\boldsymbol{T}$ are the only unknowns. It is easy to solve for ω by taking the determinant of both sides

$$\omega^3 \det(\boldsymbol{K}) = \det(\boldsymbol{T}) \det(\boldsymbol{H}) \tag{102}$$

and, since $\det(\boldsymbol{T}) = 1$, we find

$$\omega = \sqrt[3]{\frac{\det(\boldsymbol{H})}{\det(\boldsymbol{K})}} \tag{103}$$

Moreover, since $\det(\boldsymbol{K}) = d_x d_y > 0$, we observe that

$$\text{sign}[\omega] = \text{sign}[\det(\boldsymbol{H})] \tag{104}$$

such that

$$\omega = \text{sign}[\det(\boldsymbol{H})] \, \|\omega\| \tag{105}$$

Keeping track of the sign is important since the scaling $\omega \neq 0$ is arbitrary and it could change the sign of $\boldsymbol{H}$ as compared to $\boldsymbol{K}\boldsymbol{T}$ [see Eq. (79)]. However, as will soon become apparent, actual computation of ω is never required.

Proceed, therefore, by constructing the orthogonal Procrustes problem loss function from the equality in Eq. (101),

$$\min_{\boldsymbol{T} \in \text{SO}(3)} \; L(\boldsymbol{T}) = \frac{1}{2} \| \omega \boldsymbol{K}^T - \boldsymbol{T} \boldsymbol{H}^T \|_F^2 \tag{106}$$

Following the usual approach, compute the intermediate matrix

$$\boldsymbol{G} = \omega \boldsymbol{K}^T \boldsymbol{H} \tag{107}$$

and then find the singular value decomposition

$$\boldsymbol{G} = \boldsymbol{U} (\|\omega\| \boldsymbol{S}) \boldsymbol{V}^T \tag{108}$$

The optimal solution for $\boldsymbol{T}$ is once again given by [exact same form as in Eq. (39)]

$$\hat{\boldsymbol{T}} = \boldsymbol{U} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \det(\boldsymbol{U})\det(\boldsymbol{V}) \end{bmatrix} \boldsymbol{V}^T \tag{109}$$

where the term $\det(\boldsymbol{U})\det(\boldsymbol{V})$ guarantees that the attitude transformation matrix is a proper orthogonal matrix [i.e., that $\det(\boldsymbol{T}) = +1$] and describes a right-handed coordinate system. We almost always find $\det(\boldsymbol{U})\det(\boldsymbol{V}) = +1$ in practice due to the structure of $\boldsymbol{K}$, though explicit computation of $\det(\boldsymbol{U})\det(\boldsymbol{V})$ is required to formally *guarantee* that $\hat{\boldsymbol{T}}$ is proper orthogonal.

To avoid needing to compute ω, define the matrix $\boldsymbol{G}'$

$$\boldsymbol{G}' = \text{sign}[\det(\boldsymbol{H})] \, \boldsymbol{K}^T \boldsymbol{H} \tag{110}$$

such that $\boldsymbol{G} = \|\omega\| \boldsymbol{G}'$. Therefore, letting the SVD of $\boldsymbol{G}'$ be

$$\boldsymbol{G}' = \boldsymbol{U} \boldsymbol{S} \boldsymbol{V}^T \tag{111}$$

we observe that

$$\boldsymbol{G} = \|\omega\| \boldsymbol{G}' = \|\omega\| \boldsymbol{U} \boldsymbol{S} \boldsymbol{V}^T = \boldsymbol{U} (\|\omega\| \boldsymbol{S}) \boldsymbol{V}^T \tag{112}$$

which is exactly the same as Eq. (108). Thus, we may compute the SVD of the intermediate matrix $\boldsymbol{G}'$ instead of $\boldsymbol{G}$ since the SVD of both produces the same $\boldsymbol{U}$ and $\boldsymbol{V}$—and, therefore, both produce the same attitude estimate $\hat{\boldsymbol{T}}$ in Eq. (109). Computing the SVD of $\boldsymbol{G}'$ from Eq. (110) instead of the SVD of $\boldsymbol{G}$ from Eq. (107) allows us to avoid computing ω from Eq. (103).

5) ERROR COVARIANCE FOR CALIBRATION PARAMETERS

Analytic construction of the error-covariance expressions for all parameters is straightforward but tedious. The key results are summarized here, with detailed partial derivatives shown in the appendix. Details of these computations are left as an exercise to the reader.

The objective here is to derive analytic expressions for the error-covaraince of the five calibration parameters shown in Eq. (99). This is accomplished by first determining the error-covariances of $\boldsymbol{h}$, $\boldsymbol{V}_h$, and $\boldsymbol{b}$. Equation (73) is related to the negative log-likelihood. Thus, the error-covariance of $\boldsymbol{h}$ to within first-order can be shown to be given by [67]

$$\boldsymbol{P}_h \equiv \text{cov}\{\boldsymbol{h}\} = \left(\sum_{i=1}^{n} \boldsymbol{D}_i^T \boldsymbol{Q}_i^{-1} \boldsymbol{D}_i \right)^{-1} \tag{113}$$

where $\boldsymbol{D}_i$ is from Eq. (50) and $\boldsymbol{Q}_i$ is from Eq. (71), with both being evaluated evaluated using the true values of $[u_i, v_i]$ and $\boldsymbol{h}$. However, the measurements or estimates must be used in practice to compute $\boldsymbol{P}_h$, which leads to second-order error effects.

The derivation of $\boldsymbol{P}_v \equiv \text{cov}\{\text{vec}(\boldsymbol{V}_h)\}$ is accomplished using a standard covariance transformation, given by taking partials of the quantities within $\boldsymbol{V}_h$ from Eq. (95). We briefly observe that χ^2 terms are also present [which should be evident from Eq. (86)], but these terms are ignored because they are typically much smaller in magnitude than the first-order Gaussian terms.

Note that $\boldsymbol{h}$ is defined as $\text{vec}(\boldsymbol{H}^T)$ while the columns of $\boldsymbol{H}$ are used to define $\boldsymbol{h}_i$ and $\boldsymbol{h}_j$ in Eq. (85). Thus, the error-covariance of $\text{vec}(\boldsymbol{H})$ is required. This is given by

$$\bar{\boldsymbol{P}}_h \equiv \text{cov}\{\text{vec}(\boldsymbol{H})\} = \bar{\boldsymbol{K}}_{3,3}^T \boldsymbol{P}_h \bar{\boldsymbol{K}}_{3,3} \tag{114}$$

where $\bar{\boldsymbol{K}}_{3,3}$ is 9×9 commutation matrix [71], given by

$$\bar{\boldsymbol{K}}_{3,3} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{115}$$

where we also observe that $\bar{\boldsymbol{K}}_{3,3}^{-1} = \bar{\boldsymbol{K}}_{3,3}^{T}$. Now, partition the 9×9 matrix $\bar{\boldsymbol{P}}_h$ as computed from Eq. (114) into 3×3 matrix sub-blocks:

$$\bar{\boldsymbol{P}}_h = \begin{bmatrix} \bar{\boldsymbol{P}}_{h_{11}} & \bar{\boldsymbol{P}}_{h_{12}} & \bar{\boldsymbol{P}}_{h_{13}} \\ \bar{\boldsymbol{P}}_{h_{12}}^T & \bar{\boldsymbol{P}}_{h_{22}} & \bar{\boldsymbol{P}}_{h_{23}} \\ \bar{\boldsymbol{P}}_{h_{13}}^T & \bar{\boldsymbol{P}}_{h_{23}}^T & \bar{\boldsymbol{P}}_{h_{33}} \end{bmatrix} \tag{116}$$

The nine sub-blocks of $\bar{\boldsymbol{P}}_h$ may be used to compute the elements of the 36×36 matrix $\boldsymbol{P}_v \equiv \text{cov}\{\text{vec}(\boldsymbol{V}_h)\}$. These may be found analytically using the equations provided in the appendix. For convenience, the matrix $\boldsymbol{P}_v$ is partitioned into into 36 elements (each being its own 6×6 matrix, with only 21 being unique) as follows:

$$\boldsymbol{P}_v \equiv \begin{bmatrix} \boldsymbol{P}_{v_{11}} & \boldsymbol{P}_{v_{12}} & \boldsymbol{P}_{v_{13}} & \boldsymbol{P}_{v_{14}} & \boldsymbol{P}_{v_{15}} & \boldsymbol{P}_{v_{16}} \\ \boldsymbol{P}_{v_{12}}^T & \boldsymbol{P}_{v_{22}} & \boldsymbol{P}_{v_{23}} & \boldsymbol{P}_{v_{24}} & \boldsymbol{P}_{v_{25}} & \boldsymbol{P}_{v_{26}} \\ \boldsymbol{P}_{v_{13}}^T & \boldsymbol{P}_{v_{23}}^T & \boldsymbol{P}_{v_{33}} & \boldsymbol{P}_{v_{34}} & \boldsymbol{P}_{v_{35}} & \boldsymbol{P}_{v_{36}} \\ \boldsymbol{P}_{v_{14}}^T & \boldsymbol{P}_{v_{24}}^T & \boldsymbol{P}_{v_{34}}^T & \boldsymbol{P}_{v_{44}} & \boldsymbol{P}_{v_{45}} & \boldsymbol{P}_{v_{46}} \\ \boldsymbol{P}_{v_{15}}^T & \boldsymbol{P}_{v_{25}}^T & \boldsymbol{P}_{v_{35}}^T & \boldsymbol{P}_{v_{45}}^T & \boldsymbol{P}_{v_{55}} & \boldsymbol{P}_{v_{56}} \\ \boldsymbol{P}_{v_{16}}^T & \boldsymbol{P}_{v_{26}}^T & \boldsymbol{P}_{v_{36}}^T & \boldsymbol{P}_{v_{46}}^T & \boldsymbol{P}_{v_{56}}^T & \boldsymbol{P}_{v_{66}} \end{bmatrix} \tag{117}$$

Closed-form expressions for the 21 unique elements are shown in the appendix.

The error-covariance of $\boldsymbol{b}$ now can be derived in a similar way that the error-covariance of $\boldsymbol{h}$ has been derived

$$\begin{aligned} \boldsymbol{P}_b &\equiv \text{cov}\{\boldsymbol{b}\} \\ &= \left\{ \boldsymbol{V}_h^T \left[(\boldsymbol{I}_6 \otimes \boldsymbol{b}^T) \boldsymbol{P}_v (\boldsymbol{I}_6 \otimes \boldsymbol{b}) \right]^{-1} \boldsymbol{V}_h \right\}^{-1} \end{aligned} \tag{118}$$

where $\boldsymbol{V}_h$ is as defined in Eq. (95).

The remaining step is to transform $\boldsymbol{P}_b$ into the covariance of the five calibration parameters, $\boldsymbol{P}_c$. Therefore, proceed by defining the calibration parameter vector $\boldsymbol{x}_c$ as

$$\boldsymbol{x}_c \equiv [d_x \;\; \alpha \;\; d_y \;\; u_p \;\; v_p]^T \tag{119}$$

The error-covariance of $\boldsymbol{x}_c$ is given by

$$\boldsymbol{P}_c \equiv \text{cov}\{\boldsymbol{x}_c\} = \left(\frac{\partial \boldsymbol{x}_c}{\partial \boldsymbol{b}} \right) \boldsymbol{P}_b \left(\frac{\partial \boldsymbol{x}_c}{\partial \boldsymbol{b}} \right)^T \tag{120}$$

The 30 partials making up the 5×6 matrix $\partial \boldsymbol{x}_c / \partial \boldsymbol{b}$ may be computed analytically and are shown in the appendix.

C. UNCALIBRATED CAMERA (ONLY UNKNOWN FOCAL LENGTH)

In many cases, it is not desirable to estimate all five of the camera calibration parameters. The principal point coordinates are poorly observable and highly correlated with the attitude estimate [28]. Consequently, the usual approach is to fix the principal point coordinate and treat this as a sensor misalignment (bias state in the navigation filter). For well-built focal plane arrays, it is also generally reasonable to assume the pixels are square ($d_x = d_y$) and that the pixel columns are orthogonal to the pixel rows (no skewness, $\alpha = 0$). In this case, only the ratio of the focal length to pixel pitch is unknown.

1) LEAST SQUARES SOLUTION

The overall approach is identical to the general uncalibrated case, but some important simplifications may be made. Specifically, first observe that $\boldsymbol{B}$ simplifies to

$$\boldsymbol{B} = \boldsymbol{K}^{-T} \boldsymbol{K}^{-1} = \frac{1}{d_x^2} \begin{bmatrix} 1 & 0 & -u_p \\ 0 & 1 & -v_p \\ -u_p & -v_p & d_x^2 + u_p^2 + v_p^2 \end{bmatrix} \tag{121}$$

where everything is known except d_x. Therefore, using the same definition of $\boldsymbol{b}$ as before,

$$\boldsymbol{b} = \frac{1}{d_x^2} \begin{bmatrix} 1 \\ 0 \\ 1 \\ -u_p \\ -v_p \\ d_x^2 + u_p^2 + v_p^2 \end{bmatrix} \tag{122}$$

Then, using Eq. (86),

$$\begin{aligned} \boldsymbol{v}_{ij}^T \boldsymbol{b} = \frac{1}{d_x^2} \big[& h_{1j} h_{1i} + h_{2j} h_{2i} \\ & - \left(h_{1j} h_{3i} + h_{3j} h_{1i} \right) u_p - \left(h_{2j} h_{3i} + h_{3j} h_{2i} \right) v_p \\ & + h_{3j} h_{3i} \left(d_x^2 + u_p^2 + v_p^2 \right) \big] = 0 \end{aligned} \tag{123}$$

which may be rearranged to

$$\begin{aligned} h_{3j} h_{3i} \, c = & \left(h_{1j} h_{3i} + h_{3j} h_{1i} \right) u_p \\ & + \left(h_{2j} h_{3i} + h_{3j} h_{2i} \right) v_p - h_{1j} h_{1i} - h_{2j} h_{2i} \end{aligned} \tag{124}$$

where we have introduced the intermediate variable c

$$c = d_x^2 + u_p^2 + v_p^2 \tag{125}$$

Letting the right-hand side be y_{ij}

$$\begin{aligned} y_{ij} = & \left(h_{1j} h_{3i} + h_{3j} h_{1i} \right) u_p + \left(h_{2j} h_{3i} + h_{3j} h_{2i} \right) v_p \\ & - h_{1j} h_{1i} - h_{2j} h_{2i} \end{aligned} \tag{126}$$

Therefore, stack the same six relationships as in Eq. (95) but where c is the only unknown,

$$\begin{bmatrix} h_{32}h_{31} \\ h_{33}h_{31} \\ h_{33}h_{32} \\ h_{31}^2 - h_{32}^2 \\ h_{31}^2 - h_{33}^2 \\ h_{32}^2 - h_{33}^2 \end{bmatrix} c = \begin{bmatrix} y_{12} \\ y_{13} \\ y_{23} \\ y_{11} - y_{22} \\ y_{11} - y_{33} \\ y_{22} - y_{33} \end{bmatrix} \rightarrow \boldsymbol{g}c = \boldsymbol{y} \tag{127}$$

which may be solved for the scalar c in either the least squares sense or the TLS sense [67]. Therefore, with c known, the unknown d_x may be found by rearranging Eq. (125),

$$\hat{d}_x = \sqrt{\hat{c} - u_p^2 - v_p^2} \tag{128}$$

This solution for d_x is exact and make no assumptions beyond the pinhole camera model.

2) ERROR COVARIANCE FOR d_x

It is possible to analytically compute the error-variance for $\hat{d}_x$. The details are now shown.

To begin, compute the error-covariance of $\boldsymbol{h}_{gy} \equiv \text{vec}([\boldsymbol{g} \;\; \boldsymbol{y}])$, which is given by

$$\boldsymbol{P}_{gy} \equiv \text{cov}\{\text{vec}(\boldsymbol{h}_{gy})\} = \begin{bmatrix} \boldsymbol{Z}_{gg}\boldsymbol{P}_h\boldsymbol{Z}_{gg}^T & \boldsymbol{Z}_{gg}\boldsymbol{P}_h\boldsymbol{Z}_{yy}^T \\ \boldsymbol{Z}_{yy}\boldsymbol{P}_h\boldsymbol{Z}_{gg}^T & \boldsymbol{Z}_{yy}\boldsymbol{P}_h\boldsymbol{Z}_{yy}^T \end{bmatrix} \tag{129}$$

The matrix partials $\boldsymbol{Z}_{gg}$ and $\boldsymbol{Z}_{yy}$ are analytic functions of $\boldsymbol{H}$ and $\{\boldsymbol{u}\}_{i=1}^n$ are given in the appendix.

Defining the vector $\boldsymbol{z}_c \equiv [c \;\; -1]^T$, the error-variance of $\hat{c}$ is given by

$$p_c \equiv \text{var}\{\hat{c}\} = \boldsymbol{h}_h^T[(\boldsymbol{z}_c^T \otimes \boldsymbol{I}_6)\boldsymbol{P}_{hy}(\boldsymbol{z}_c \otimes \boldsymbol{I}_6)]^{-1}\boldsymbol{h}_h \tag{130}$$

It follows from Eq. (128), therefore, that the error-variance of $\hat{d}_x$ is given by

$$\text{var}\{\hat{d}_x\} = \frac{p_c}{4(c - u_p^2 - v_p^2)} \tag{131}$$

where all quantities are evaluated at their respective true value in theory, but are replaced by their estimates in practice. Note that the matrix $(\boldsymbol{z}_c^T \otimes \boldsymbol{I}_6)\boldsymbol{P}_{hy}(\boldsymbol{z}_c \otimes \boldsymbol{I}_6)$ in Eq. (130) may be singular. In practice, the pseudo-inverse works well but should be checked using Monte Carlo simulations. This singular condition may cause issues in computing the TLS estimates of c, which may be overcome by using the approach shown in Ref. [72]. Fortunately, a simple linear least-squares solution is almost always adequate, but this should be validated for any important application by a comparison with the more statistically rigorous TLS solution.

VI. NUMERICAL RESULTS

A. VALIDATION OF INVARIANTS FOR ASTERISM DESCRIPTORS

The invariance of the asterism descriptors proposed in this work may be demonstrated with data from a real starfield image. Therefore, as an example, consider the image shown in Fig. 8. Using the pixel coordinates of the five labeled stars (see Table 3,) we may compute some of the invariants discussed in Section. III.

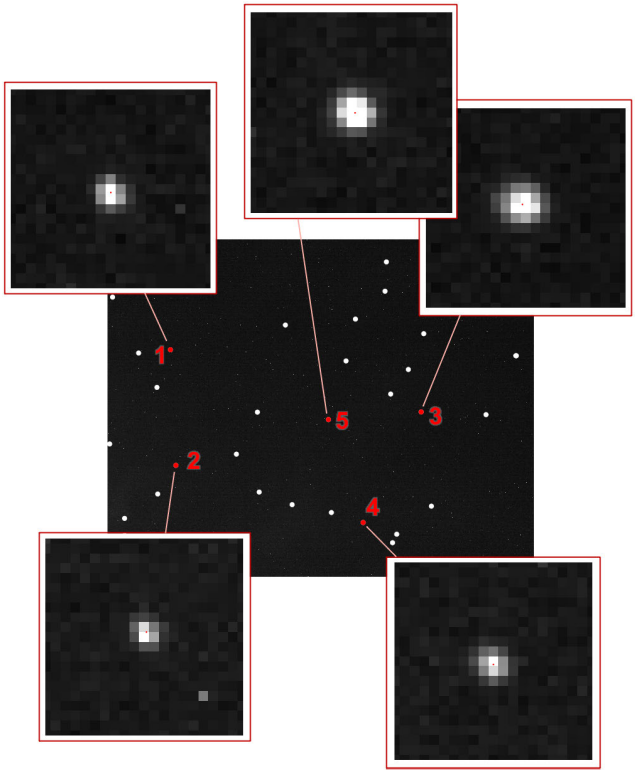

FIGURE 8. Example starfield image collected in August 2019 near Troy, NY by RPI student Grace Quintero. The image has a 2, 048 × 2, 592 pixel resolution and the camera has $d_x = d_y = 7,250$ (computed from vendor specifications; calibrated value is different). White dots have been placed on top of apparent stars to make their locations visible. Five example stars are chosen at random and are marked with red dots, with a zoom-in view provided for each.

TABLE 3. Measured star centroid coordinates (subpixel accuracy) in the image for the five stars marked in red in Fig. 8. Pixel coordinates $[u_i, v_i]$ use are measured from the image upper left-hand corner.

Star #	u_i (pixel) *column coord.*	v_i (pixel) *row coord.*
1	382.00	668.35
2	415.23	1,371.51
3	1,907.69	1,046.64
4	1,555.00	1,719.33
5	1,343.35	1,093.04

For the case of a calibrated camera ($\boldsymbol{K}$ known), we may compute the line-of-sight unit vectors $\boldsymbol{a}_i$ to the five reference stars in Fig. 8 using Eq. (13). These unit vectors $\{\boldsymbol{a}_i\}_{i=1}^5$ may be used to find a number of different invariants directly from the image data. For example, using just the first three stars it is possible to compute the inter-star angles $\theta_{12}, \theta_{23}, \theta_{31}$ [see Eq. (14)], the dihedral angles $\Theta_1, \Theta_2, \Theta_3$ [see Eqs. (18)–(20)], or the permutation invariants F_1, F_2, F_3 [see Eqs. (21)–(23)]. We could compute these invariants for different combinations of stars (e.g., any set of $10 - 3 = 7$ independent inter-star angles for the highlighted five-star set), but doing so provides no additional insight. Numerical

TABLE 4. Example asterism invariants for a calibrated camera as computed from stars 1–3 from example starfield image shown in Fig. 8.

Invariant	From Star Catalog (deg)	From Image (deg) *Specification*	From Image (deg) *Self-calibration*
θ_{12}	5.4891	5.4711	5.4926
θ_{23}	11.9641	11.9461	11.9611
θ_{31}	12.2993	12.3038	12.2995
Θ_1	73.7269	73.5240	73.6877
Θ_2	80.5477	80.8286	80.5683
Θ_3	26.2928	26.2124	26.3116
F_1	29.7526	29.7211	29.7532
F_2	-13.1838	-13.1927	-13.1723
F_3	-1.7373	-1.8533	-1.7536

values for these invariants are shown in Table 4, where values have been computed directly from the Hipparcos catalog (left column), with $\boldsymbol{K}$ directly from the camera and lens specifications (center column), and with $\boldsymbol{K}$ computed using the self-calibration algorithm from Section V-B (right column).

A few important observations may now be made. First, we observe the numerical values for any given invariant is nearly the same in all three columns. They are not exactly the same since there is measurement noise associated with the observed star locations in the real image. This demonstrates that the quantities listed in Table 4 are indeed invariants, thus allowing us to query an index built from the star catalog using the numerical values computed from a calibrated image of unknown attitude. Second, the analyst could choose any of these invariants to build a descriptor. However, since an asterism of d stars has $2d-3$ independent invariants, only three ($6-3=3$) of the invariants in Table 4 can be simultaneously independent. Third, the reader may verify to themselves that the invariants F_1, F_2, F_3 are permutation invariant by substitution into Eqs. (21)–(23). For example, using the inter-star values form the Hipparcos catalog,

$$\begin{aligned} &F_1(5.4891, 11.9641, 12.2993) \\ &\quad = F_1(11.9641, 12.2993, 5.4891) \\ &\quad = F_1(12.2993, 5.4891, 11.9641) \\ &\quad = 29.7526 \end{aligned}$$

A similar exercise may be performed for F_2 and F_3.

The same example image (see Fig. 8) may also be used to validate the invariants for an uncalibrated camera. Here, we focus on the invariants that use determinants and cross ratios for a five-star asterism [e.g., Eq. (29)], as these offer the clearest understanding of the problem. The numerical values for some of the possible invariants are summarized in Table 5. Once again, we observe the numerical values of any given invariant is nearly the same when computed form the Hipparcos catalog or from the observed star centroids.

Unlike the calibrated camera results, the matrix $\boldsymbol{K}$ is not known for an uncalibrated camera. As a result, we cannot compute $\boldsymbol{a}_i$ from Eq. (13). We remind the reader that the invariant values in the right-hand column of Table 5 (labeled "From Image") are computed directly from the star image coordinates in units of pixels (exactly as they appear in

TABLE 5. Example asterism invariants for an uncalibrated camera as computed from stars 1–5 from example starfield image shown in Fig. 8.

Invariant	From Star Catalog	From Image
$I_1^{(a)} = \mathrm{Cr}(\ell_{12}, \ell_{13}, \ell_{14}, \ell_{15})$	-2.4029	-2.4184
$I_1^{(b)} = \mathrm{Cr}(\ell_{15}, \ell_{13}, \ell_{14}, \ell_{12})$	-0.4162	-0.4135
$I_1^{(c)} = \mathrm{Cr}(\ell_{13}, \ell_{14}, \ell_{15}, \ell_{12})$	3.4029	3.4184
$I_1^{(d)} = \mathrm{Cr}(\ell_{14}, \ell_{12}, \ell_{13}, \ell_{15})$	0.2939	0.2925
$I_1^{(e)} = \mathrm{Cr}(\ell_{13}, \ell_{12}, \ell_{14}, \ell_{15})$	0.7061	0.7075
$I_1^{(f)} = \mathrm{Cr}(\ell_{14}, \ell_{13}, \ell_{15}, \ell_{12})$	1.4162	1.4135
$I_2 = \mathrm{Cr}(\ell_{21}, \ell_{23}, \ell_{24}, \ell_{25})$	7.6483	7.5325
$I_3 = \mathrm{Cr}(\ell_{31}, \ell_{32}, \ell_{34}, \ell_{35})$	2.9537	2.9110
$I_4 = \mathrm{Cr}(\ell_{41}, \ell_{42}, \ell_{43}, \ell_{45})$	0.3862	0.3865
$I_5 = \mathrm{Cr}(\ell_{51}, \ell_{52}, \ell_{53}, \ell_{54})$	1.3142	1.3211
$J(I_1)$	11.5589	11.6447
$J(I_2)$	53.9078	52.2672
$J(I_3)$	9.3205	9.1344
$J(I_4)$	7.9036	7.8977
$J(I_5)$	16.5448	16.0558
$J'(I_1)$	2.3505	2.3470
$J'(I_2)$	2.0589	2.0609
$J'(I_3)$	2.4746	2.4890
$J'(I_4)$	2.6118	2.6125
$J'(I_5)$	2.2215	2.2298

Table 3). That is, letting $\bar{\boldsymbol{u}}_i^T = [u_i, v_i, 1]$, we compute the invariants using Eqs. (26), (27), and (29). Since the invariants are computed directly from the image coordinates, we may find them without any *a priori* calibration knowledge—thus allowing recognition of a star pattern with an image from an arbitrary projective camera having an unknown calibration.

The first section of Table 5 lists the six possible invariants that result from using star #1 in Fig. 8 as the reference star. As discussed in Section III-C4, while there are $4! = 24$ possible permutations of the remaining four points, these produce only six different invariant values. We observe that the six numerical values in this example do indeed follow the classical relations expected from Eq. (30). For example, using the star-catalog invariant values from Table 5,

$$\begin{aligned} I_1^{(a)} &= \tau = -2.4029 \\ I_1^{(b)} &= \tau^{-1} = -0.4162 \\ I_1^{(c)} &= 1 - \tau = 3.4029 \\ I_1^{(d)} &= (1-\tau)^{-1} = 0.2939 \\ I_1^{(e)} &= \tau(\tau-1)^{-1} = 0.7061 \\ I_1^{(f)} &= \tau^{-1}(\tau-1) = 1.4162 \end{aligned}$$

A similar expansion of possibilities could be carried out for the invariants listed in the second section of Table 5. This is not done here for the sake of brevity.

Since the six cross-ratio values arising from a particular choice of reference star are not independent, we may instead compute a single *unique* invariant for each case. We present two possibilities in this manuscript, including $J(\cdot)$ in Eq. (31) and $J'(\cdot)$ in Eq. (32). Again using I_1 as the example, we observe that J and J' are unchanged by permutations in

assignment of the non-reference star:

$$J(I_1^{(a)}) = J(I_1^{(b)}) = J(I_1^{(c)}) = J(I_1^{(d)}) = J(I_1^{(e)}) = J(I_1^{(f)}) = 11.5589$$

and

$$J'(I_1^{(a)}) = J'(I_1^{(b)}) = J'(I_1^{(c)}) = J'(I_1^{(d)}) = J'(I_1^{(e)}) = J'(I_1^{(f)}) = 2.3505$$

Similar results may be shown for I_2, I_3, I_4, I_5 and their permutation invariants J_2, J_3, J_4, J_5 and J_2', J_3', J_4', J_5'. Computation of all the remaining possibilities is trivial from the data provided in Table 3 and Table 5. Such an exercise is left to the reader.

As with the star triad, the navigation analyst could construct an asterism descriptor using any combination of independent invariants from Table 5. There are also other invariants that could also be used in conjunction with (or in place of) the invariants Table 5—such as the coordinates of stars in a canonical frame (see Section III-C3)—so long as the combination of invariants remain independent. Recall there are $2d - 8$ independent invariants for a generic calibrated camera, allowing for only a two-element descriptor for a five-star asterism.

B. VALIDATION OF ATTITUDE DETERMINATION AND CALIBRATION

The attitude determination algorithms presented in this paper are validated through a number of numerical studies. We first show the approximate equivalence between the DLT and GTLS solution for estimating the homography matrix of an uncalibrated camera. We then present a comparative study of the attitude determination and camera self-calibration performance that may be achieved under different conditions and with different algorithms. All results in this section are performed on simulated data, since computing meaningful attitude determination and camera calibration performance statistics is difficult on real images due a lack of accurate ground truth knowledge.

1) DLT VS. GTLS FOR AN UNCALIBRATED CAMERA

Two non-iterative algorithms are presented for finding the homography matrix from a single starfield image, thus allowing for simultaneous camera calibration (all five camera parameters) and attitude determination. The first method is the direct linear transform (DLT) presented in Sec. V-B1, which eliminates iteration by solving an unweighted least squares problem. The second method is the generalized total least squares (GTLS) solution presented in Sec. V-B3, which eliminates iteration by assuming a narrow FOV and that each row of $\boldsymbol{D}$ has the same covariance.

For most realistic camera configurations, these two methods produce nearly equivalent estimates of $\boldsymbol{H}$. The two methods, therefore, also produce nearly equivalent estimates of the five camera calibration parameters and camera attitude. This may be easily illustrated by a numerical example.

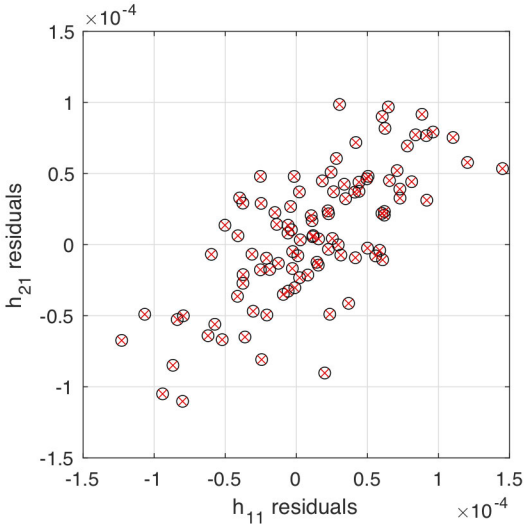

FIGURE 9. Comparison of residuals in h_{11} and h_{12} for the DLT estimate (red ×) and the GTLS estimate (black circles).

Consider a camera with a 20 deg FOV and a $1,024 \times 1,024$ pixel focal plane array. Assuming a star centroiding error of 0.1 pixel, we may compute the errors in the nine elements of $\boldsymbol{h}$ (assuming the normalization $\|\boldsymbol{h}\| = 1$). A representative scatter plot of such residuals is shown in Fig. 9 for 100 runs of a Monte Carlo, where we can see the estimates from the DLT are indiscernible from the estimate from the GTLS. This same trend holds for scatter plots of all elements of $\boldsymbol{H}$.

This same example scenario may be used to validate the analytic covariance expressions. Specifically, consider the numerically computed covariance of the five calibration parameters from 50,000 Monte Carlo runs

$$\boldsymbol{P}_c^{(num)} = \begin{bmatrix} 0.064 & -0.002 & 0.001 & 0.123 & -0.100 \\ -0.002 & 0.113 & 0.004 & -0.239 & -0.003 \\ 0.001 & 0.004 & 0.045 & 0.053 & 0.021 \\ 0.123 & -0.239 & 0.053 & 3.541 & 0.487 \\ -0.100 & -0.003 & 0.021 & 0.487 & 2.740 \end{bmatrix}$$

The difference in the numerical covariance for DLT and the GTLS is on the order 10^{-6} or less (at least three decimal places past what is reported above). The numerical covariance compares favorably to the analytic covariance computed using Eq. (120)

$$\boldsymbol{P}_c^{(comp)} = \begin{bmatrix} 0.064 & -0.002 & 0.001 & 0.123 & -0.105 \\ -0.002 & 0.114 & 0.005 & -0.235 & 0.001 \\ 0.001 & 0.005 & 0.045 & 0.053 & 0.020 \\ 0.123 & -0.235 & 0.053 & 3.513 & 0.488 \\ -0.105 & 0.001 & 0.020 & 0.488 & 2.788 \end{bmatrix}$$

Other example scenarios and Monte Carlo runs also show good agreement between analytic and numerical error covariances.

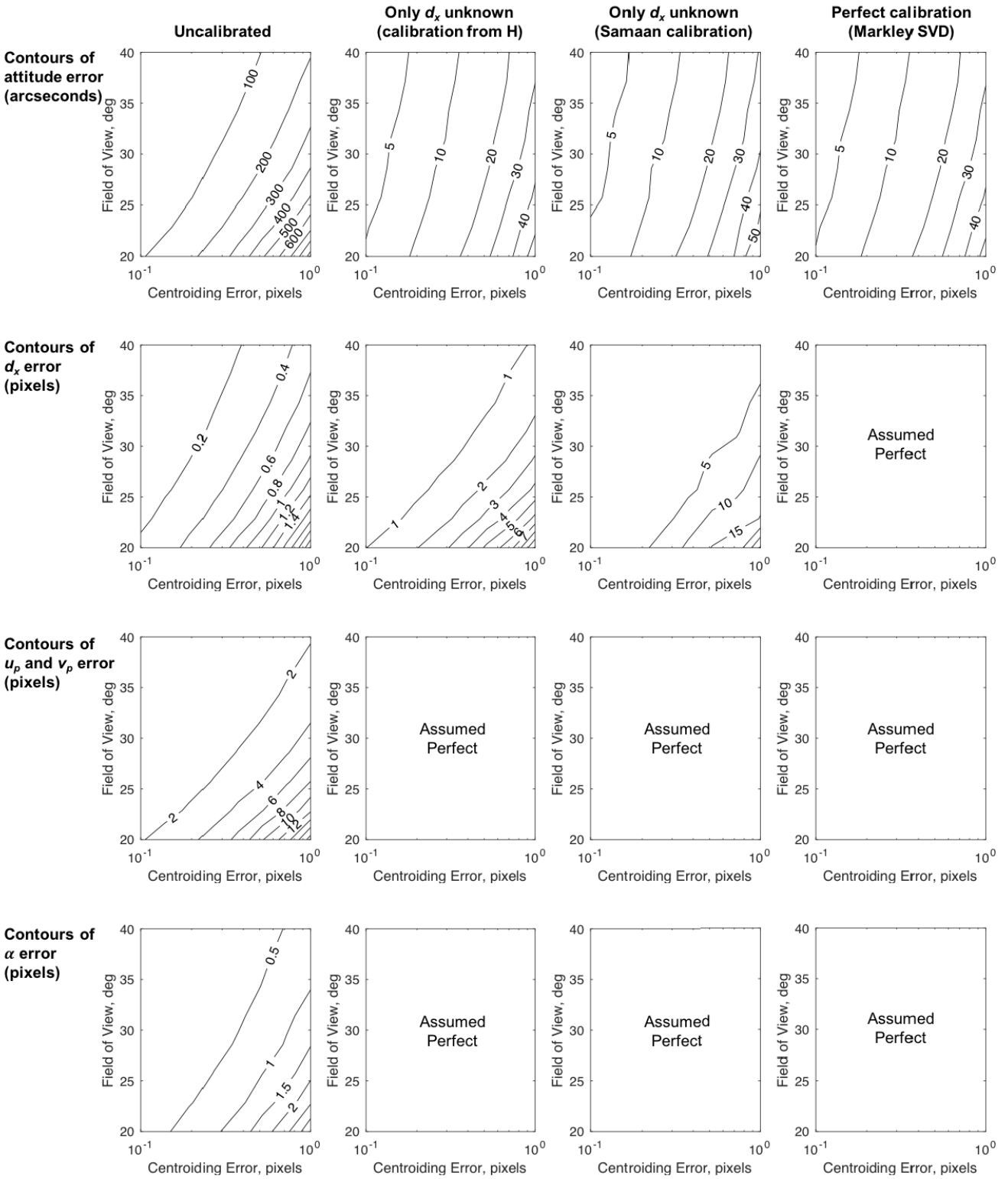

FIGURE 10. Contours of attitude determination (top row) and calibration performance (bottom three rows) for a variety of camera configurations. Left column assumes that no calibration information is available (algorithm from Sec. V-B). The middle two columns assume that d_x is the only unknown calibration parameter, with the second column using the algorithm from Sec. V-C and the third column using the algorithm from [15]. The right column assumes a perfectly calibrated camera and estimates the attitude using the algorithm from [12]. All results show standard deviation for the specified parameter as computed from a 2,000-run Monte Carlo at each combination of camera FOV and star centroiding error.

2) CALIBRATION AND ATTITUDE DETERMINATION PERFORMANCE COMPARISONS

To assess the attitude determination performance, a simple Monte Carlo analysis was performed where the camera FOV was varied from 20 deg to 40 deg and the star centroiding error was varied from 0.1 pixels to 1.0 pixel. At each combination of FOV and centroiding error, a 2,000-run Monte Carlo simulation was performed and the results are summarized in Fig. 10.

The simulation results in Fig. 10 assume a camera with a $2,048 \times 2,048$ pixel focal plane array (a 4.2 megapixel image) that is capable of viewing stars of magnitude 5.5 or brighter. A camera with a different resolution or with a different star magnitude threshold will have different performance. For each run of the Monte Carlo for each design condition, the camera was randomly pointed with a uniform distribution around the celestial sphere. Star directions were obtained from the Hipparcos star catalog.

The results of Fig. 10 are exactly as we would expect. First, looking at the top row, the best attitude determination performance is achieved with perfect *a priori* calibration knowledge and the worst attitude determination performance is achieved in the absence of any *a priori* calibration knowledge. The primary cause of the larger attitude error for a completely uncalibrated camera is due to the relatively poor observability of the principal point coordinates $[u_p, v_p]$, which was discussed extensively in Ref. [28]. If we assume the principal point coordinates $[u_p, v_p]$ are known, that the pixels are square ($d_x = d_y$), and that the image rows and columns are perfectly orthogonal ($\alpha = 0$), then d_x is the only unknown (that is, if only the ratio of the focal length to pixel pitch is unknown). In this case, the method we introduce in Section V-C marginally outperforms the method proposed by Samaan, *et al.*, in Ref. [15] in estimating d_x (see second row of Fig. 10). Again looking at the top row of Fig. 10, either method (ours or from Ref. [15]) is capable of producing attitude determination performance that is very close to the perfectly calibrated camera—with the performance of our method being nearly indistinguishable from that of a perfectly calibrated camera.

VII. CONCLUSION

Digital images of stars collected by projective cameras contain a wealth of information that is useful for many scientific and engineering applications. In this work, it is shown how invariant theory may be used to recognize the apparent pattern of stars in an image for a number of important cases: generic calibrated cameras, narrow field-of-view (FOV) calibrated cameras, generic uncalibrated cameras, and narrow FOV uncalibrated cameras. For each case, the number of independent invariants is enumerated and examples of easily computable invariants are presented. This provides the reader with an easy-to-implement guide for recognizing star patterns (i.e., asterisms) for a wide array of different imaging systems.

With the star identification problem solved, we explore how matched stars may be used to simultaneously calibrate a camera and compute the attitude. A compact algorithm is provided that allows for calibration with a single star image, in contrast to many past algorithms that require an ensemble of star images. Building on this framework, a simplified algorithm is shown for the important (and quite common) special case where only the ratio of focal length to pixel pitch ($d_x = f/\mu_x$) is unknown. Our algorithm for simultaneously finding d_x and camera attitude provides virtually the same attitude determination performance as a perfectly calibrated camera.

APPENDIX

A. ANALYTIC EXPRESSIONS FOR ELEMENTS OF P_v

It is possible to compute the partial derivatives of Eq. (86) to analytically find the 36 elements of the covariance matrix $\boldsymbol{P}_v$ from Eq. (117). To simplify notation, we first introduce the duplication matrix $\boldsymbol{C}$ to account for the repeated elements in vec($\boldsymbol{B}$) that occur due to the symmetry of $\boldsymbol{B}$. Letting $\boldsymbol{Cb} =$ vec($\boldsymbol{B}$), it follows that

$$\boldsymbol{C} \equiv \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{132}$$

Therefore, after performing the requisite calculations, we find the following analytic expressions for the elements of $\boldsymbol{P}_v$ from Eq. (117):

$$\begin{aligned} \boldsymbol{P}_{v_{11}} &= \boldsymbol{C}^T \left[(\boldsymbol{h}_2 \otimes \boldsymbol{I}_3)\ (\boldsymbol{I}_3 \otimes \boldsymbol{h}_1)\right] \begin{bmatrix} \bar{\boldsymbol{P}}_{h_{11}} & \bar{\boldsymbol{P}}_{h_{12}} \\ \bar{\boldsymbol{P}}_{h_{12}}^T & \bar{\boldsymbol{P}}_{h_{22}} \end{bmatrix} \\ &\quad \times \left[(\boldsymbol{h}_2 \otimes \boldsymbol{I}_3)\ (\boldsymbol{I}_3 \otimes \boldsymbol{h}_1)\right]^T \boldsymbol{C} \end{aligned} \tag{133a}$$

$$\begin{aligned} \boldsymbol{P}_{v_{12}} &= \boldsymbol{C}^T \left[(\boldsymbol{h}_2 \otimes \boldsymbol{I}_3)\ (\boldsymbol{I}_3 \otimes \boldsymbol{h}_1)\right] \begin{bmatrix} \bar{\boldsymbol{P}}_{h_{11}} & \bar{\boldsymbol{P}}_{h_{13}} \\ \bar{\boldsymbol{P}}_{h_{12}}^T & \bar{\boldsymbol{P}}_{h_{23}} \end{bmatrix} \\ &\quad \times \left[(\boldsymbol{h}_3 \otimes \boldsymbol{I}_3)\ (\boldsymbol{I}_3 \otimes \boldsymbol{h}_1)\right]^T \boldsymbol{C} \end{aligned} \tag{133b}$$

$$\begin{aligned} \boldsymbol{P}_{v_{13}} &= \boldsymbol{C}^T \left[(\boldsymbol{h}_2 \otimes \boldsymbol{I}_3)\ (\boldsymbol{I}_3 \otimes \boldsymbol{h}_1)\right] \begin{bmatrix} \bar{\boldsymbol{P}}_{h_{12}} & \bar{\boldsymbol{P}}_{h_{13}} \\ \bar{\boldsymbol{P}}_{h_{22}} & \bar{\boldsymbol{P}}_{h_{23}} \end{bmatrix} \\ &\quad \times \left[(\boldsymbol{h}_3 \otimes \boldsymbol{I}_3)\ (\boldsymbol{I}_3 \otimes \boldsymbol{h}_2)\right]^T \boldsymbol{C} \end{aligned} \tag{133c}$$

$$\begin{aligned} \boldsymbol{P}_{v_{14}} &= \boldsymbol{C}^T \Bigg\{ \left[(\boldsymbol{h}_2 \otimes \boldsymbol{I}_3)\ (\boldsymbol{I}_3 \otimes \boldsymbol{h}_1)\right] \begin{bmatrix} \bar{\boldsymbol{P}}_{h_{11}} \\ \bar{\boldsymbol{P}}_{h_{12}}^T \end{bmatrix} \\ &\quad \times \left[(\boldsymbol{h}_1 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_1)\right]^T \\ &\quad - \left[(\boldsymbol{h}_2 \otimes \boldsymbol{I}_3)\ (\boldsymbol{I}_3 \otimes \boldsymbol{h}_1)\right] \begin{bmatrix} \bar{\boldsymbol{P}}_{h_{12}} \\ \bar{\boldsymbol{P}}_{h_{22}} \end{bmatrix} \\ &\quad \times \left[(\boldsymbol{h}_2 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_2)\right]^T \Bigg\} \boldsymbol{C} \end{aligned} \tag{133d}$$

$$\boldsymbol{P}_{v_{15}} = \boldsymbol{C}^T \Bigg\{ \left[(\boldsymbol{h}_2 \otimes \boldsymbol{I}_3)\ (\boldsymbol{I}_3 \otimes \boldsymbol{h}_1)\right] \begin{bmatrix} \bar{\boldsymbol{P}}_{h_{11}} \\ \bar{\boldsymbol{P}}_{h_{12}}^T \end{bmatrix}$$

$$\times [(\boldsymbol{h}_1 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_1)]^T - \left[(\boldsymbol{h}_2 \otimes \boldsymbol{I}_3)\ (\boldsymbol{I}_3 \otimes \boldsymbol{h}_1)\right] \begin{bmatrix} \bar{\boldsymbol{P}}_{h_{13}} \\ \bar{\boldsymbol{P}}_{h_{23}} \end{bmatrix} \times [(\boldsymbol{h}_3 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_3)]^T \Bigg\} \boldsymbol{C} \tag{133e}$$

$$\boldsymbol{P}_{v_{16}} = \boldsymbol{C}^T \Bigg\{ \left[(\boldsymbol{h}_2 \otimes \boldsymbol{I}_3)\ (\boldsymbol{I}_3 \otimes \boldsymbol{h}_1)\right] \begin{bmatrix} \bar{\boldsymbol{P}}_{h_{12}} \\ \bar{\boldsymbol{P}}_{h_{22}} \end{bmatrix} \times [(\boldsymbol{h}_2 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_2)]^T - \left[(\boldsymbol{h}_2 \otimes \boldsymbol{I}_3)\ (\boldsymbol{I}_1 \otimes \boldsymbol{h}_1)\right] \begin{bmatrix} \bar{\boldsymbol{P}}_{h_{13}} \\ \bar{\boldsymbol{P}}_{h_{22}} \end{bmatrix} \times [(\boldsymbol{h}_3 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_3)]^T \Bigg\} \boldsymbol{C} \tag{133f}$$

$$\boldsymbol{P}_{v_{22}} = \boldsymbol{C}^T \left[(\boldsymbol{h}_3 \otimes \boldsymbol{I}_3)\ (\boldsymbol{I}_3 \otimes \boldsymbol{h}_1)\right] \begin{bmatrix} \bar{\boldsymbol{P}}_{h_{11}} & \bar{\boldsymbol{P}}_{h_{13}} \\ \bar{\boldsymbol{P}}_{h_{13}}^T & \bar{\boldsymbol{P}}_{h_{33}} \end{bmatrix} \times \left[(\boldsymbol{h}_3 \otimes \boldsymbol{I}_3)\ (\boldsymbol{I}_3 \otimes \boldsymbol{h}_1)\right]^T \boldsymbol{C} \tag{133g}$$

$$\boldsymbol{P}_{v_{23}} = \boldsymbol{C}^T \left[(\boldsymbol{h}_3 \otimes \boldsymbol{I}_3)\ (\boldsymbol{I}_3 \otimes \boldsymbol{h}_1)\right] \begin{bmatrix} \bar{\boldsymbol{P}}_{h_{12}} & \bar{\boldsymbol{P}}_{h_{13}} \\ \bar{\boldsymbol{P}}_{h_{23}}^T & \bar{\boldsymbol{P}}_{h_{33}} \end{bmatrix} \times \left[(\boldsymbol{h}_3 \otimes \boldsymbol{I}_3)\ (\boldsymbol{I}_3 \otimes \boldsymbol{h}_2)\right]^T \boldsymbol{C} \tag{133h}$$

$$\boldsymbol{P}_{v_{24}} = \boldsymbol{C}^T \Bigg\{ \left[(\boldsymbol{h}_3 \otimes \boldsymbol{I}_3)\ (\boldsymbol{I}_3 \otimes \boldsymbol{h}_1)\right] \begin{bmatrix} \bar{\boldsymbol{P}}_{h_{11}} \\ \bar{\boldsymbol{P}}_{h_{13}}^T \end{bmatrix} \times [(\boldsymbol{h}_1 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_1)]^T - \left[(\boldsymbol{h}_3 \otimes \boldsymbol{I}_3)\ (\boldsymbol{I}_1 \otimes \boldsymbol{h}_1)\right] \begin{bmatrix} \bar{\boldsymbol{P}}_{h_{12}} \\ \bar{\boldsymbol{P}}_{h_{23}}^T \end{bmatrix} \times [(\boldsymbol{h}_2 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_2)]^T \Bigg\} \boldsymbol{C} \tag{133i}$$

$$\boldsymbol{P}_{v_{25}} = \boldsymbol{C}^T \Bigg\{ \left[(\boldsymbol{h}_3 \otimes \boldsymbol{I}_3)\ (\boldsymbol{I}_3 \otimes \boldsymbol{h}_1)\right] \begin{bmatrix} \bar{\boldsymbol{P}}_{h_{11}} \\ \bar{\boldsymbol{P}}_{h_{13}}^T \end{bmatrix} \times [(\boldsymbol{h}_1 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_1)]^T - \left[(\boldsymbol{h}_3 \otimes \boldsymbol{I}_3)\ (\boldsymbol{I}_1 \otimes \boldsymbol{h}_1)\right] \begin{bmatrix} \bar{\boldsymbol{P}}_{h_{13}} \\ \bar{\boldsymbol{P}}_{h_{33}} \end{bmatrix} \times [(\boldsymbol{h}_3 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_3)]^T \Bigg\} \boldsymbol{C} \tag{133j}$$

$$\boldsymbol{P}_{v_{26}} = \boldsymbol{C}^T \Bigg\{ \left[(\boldsymbol{h}_3 \otimes \boldsymbol{I}_3)\ (\boldsymbol{I}_3 \otimes \boldsymbol{h}_1)\right] \begin{bmatrix} \bar{\boldsymbol{P}}_{h_{12}} \\ \bar{\boldsymbol{P}}_{h_{23}}^T \end{bmatrix} \times [(\boldsymbol{h}_2 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_2)]^T - \left[(\boldsymbol{h}_3 \otimes \boldsymbol{I}_3)\ (\boldsymbol{I}_1 \otimes \boldsymbol{h}_1)\right] \begin{bmatrix} \bar{\boldsymbol{P}}_{h_{13}} \\ \bar{\boldsymbol{P}}_{h_{33}} \end{bmatrix} \times [(\boldsymbol{h}_3 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_3)]^T \Bigg\} \boldsymbol{C} \tag{133k}$$

$$\boldsymbol{P}_{v_{33}} = \boldsymbol{C}^T \left[(\boldsymbol{h}_3 \otimes \boldsymbol{I}_3)\ (\boldsymbol{I}_3 \otimes \boldsymbol{h}_2)\right] \begin{bmatrix} \bar{\boldsymbol{P}}_{h_{22}} & \bar{\boldsymbol{P}}_{h_{23}} \\ \bar{\boldsymbol{P}}_{h_{23}}^T & \bar{\boldsymbol{P}}_{h_{33}} \end{bmatrix} \times \left[(\boldsymbol{h}_3 \otimes \boldsymbol{I}_3)\ (\boldsymbol{I}_3 \otimes \boldsymbol{h}_2)\right]^T \boldsymbol{C} \tag{133l}$$

$$\boldsymbol{P}_{v_{34}} = \boldsymbol{C}^T \Bigg\{ \left[(\boldsymbol{h}_3 \otimes \boldsymbol{I}_3)\ (\boldsymbol{I}_3 \otimes \boldsymbol{h}_2)\right] \begin{bmatrix} \bar{\boldsymbol{P}}_{h_{12}}^T \\ \bar{\boldsymbol{P}}_{h_{13}^T} \end{bmatrix} \times [(\boldsymbol{h}_1 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_1)]^T - \left[(\boldsymbol{h}_3 \otimes \boldsymbol{I}_3)\ (\boldsymbol{I}_3 \otimes \boldsymbol{h}_2)\right] \begin{bmatrix} \bar{\boldsymbol{P}}_{h_{22}} \\ \bar{\boldsymbol{P}}_{h_{23}}^T \end{bmatrix} \times [(\boldsymbol{h}_2 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_2)]^T \Bigg\} \boldsymbol{C} \tag{133m}$$

$$\boldsymbol{P}_{v_{35}} = \boldsymbol{C}^T \Bigg\{ \left[(\boldsymbol{h}_3 \otimes \boldsymbol{I}_3)\ (\boldsymbol{I}_3 \otimes \boldsymbol{h}_2)\right] \begin{bmatrix} \bar{\boldsymbol{P}}_{h_{12}}^T \\ \bar{\boldsymbol{P}}_{h_{13}}^T \end{bmatrix} \times [(\boldsymbol{h}_1 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_1)]^T - \left[(\boldsymbol{h}_3 \otimes \boldsymbol{I}_3)\ (\boldsymbol{I}_1 \otimes \boldsymbol{h}_2)\right] \begin{bmatrix} \bar{\boldsymbol{P}}_{h_{23}} \\ \bar{\boldsymbol{P}}_{h_{33}} \end{bmatrix} \times [(\boldsymbol{h}_3 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_3)]^T \Bigg\} \boldsymbol{C} \tag{133n}$$

$$\boldsymbol{P}_{v_{36}} = \boldsymbol{C}^T \Bigg\{ \left[(\boldsymbol{h}_3 \otimes \boldsymbol{I}_3)\ (\boldsymbol{I}_3 \otimes \boldsymbol{h}_2)\right] \begin{bmatrix} \bar{\boldsymbol{P}}_{h_{22}} \\ \bar{\boldsymbol{P}}_{h_{23}}^T \end{bmatrix} \times [(\boldsymbol{h}_2 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_2)]^T - \left[(\boldsymbol{h}_3 \otimes \boldsymbol{I}_3)\ (\boldsymbol{I}_1 \otimes \boldsymbol{h}_2)\right] \begin{bmatrix} \bar{\boldsymbol{P}}_{h_{23}} \\ \bar{\boldsymbol{P}}_{h_{33}} \end{bmatrix} \times [(\boldsymbol{h}_3 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_3)]^T \Bigg\} \boldsymbol{C} \tag{133o}$$

$$\boldsymbol{P}_{v_{44}} = \boldsymbol{C}^T \Big\{ [(\boldsymbol{h}_1 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_1)] \bar{\boldsymbol{P}}_{h_{11}} \times [(\boldsymbol{h}_1 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_1)]^T + [(\boldsymbol{h}_2 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_2)] \bar{\boldsymbol{P}}_{h_{22}} \times [(\boldsymbol{h}_2 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_2)]^T - [(\boldsymbol{h}_1 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_1)] \bar{\boldsymbol{P}}_{h_{12}} \times [(\boldsymbol{h}_2 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_2)]^T - [(\boldsymbol{h}_2 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_2)] \bar{\boldsymbol{P}}_{h_{12}}^T \times [(\boldsymbol{h}_1 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_1)]^T \Big\} \boldsymbol{C} \tag{133p}$$

$$\boldsymbol{P}_{v_{45}} = \boldsymbol{C}^T \Big\{ [(\boldsymbol{h}_1 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_1)] \bar{\boldsymbol{P}}_{h_{11}} \times [(\boldsymbol{h}_1 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_1)]^T - [(\boldsymbol{h}_1 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_1)] \bar{\boldsymbol{P}}_{h_{13}} \times [(\boldsymbol{h}_3 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_3)]^T - [(\boldsymbol{h}_2 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_2)] \bar{\boldsymbol{P}}_{h_{12}}^T \times [(\boldsymbol{h}_1 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_1)]^T + [(\boldsymbol{h}_2 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_2)] \bar{\boldsymbol{P}}_{h_{23}} \times [(\boldsymbol{h}_3 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_3)]^T \Big\} \boldsymbol{C} \tag{133q}$$

$$\boldsymbol{P}_{v_{46}} = \boldsymbol{C}^T \Big\{ [(\boldsymbol{h}_1 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_1)] \bar{\boldsymbol{P}}_{h_{12}} \times [(\boldsymbol{h}_2 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_2)]^T - [(\boldsymbol{h}_1 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_1)] \bar{\boldsymbol{P}}_{h_{13}} \times [(\boldsymbol{h}_3 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_3)]^T - [(\boldsymbol{h}_2 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_2)] \bar{\boldsymbol{P}}_{h_{22}} \times [(\boldsymbol{h}_2 \otimes \boldsymbol{I}_3) + (\boldsymbol{I}_3 \otimes \boldsymbol{h}_2)]^T$$

$$+[(\boldsymbol{h}_2\otimes\boldsymbol{I}_3)+(\boldsymbol{I}_3\otimes\boldsymbol{h}_2)]\bar{\boldsymbol{P}}_{h_{23}}\times[(\boldsymbol{h}_3\otimes\boldsymbol{I}_3)+(\boldsymbol{I}_3\otimes\boldsymbol{h}_3)]^T\Big\}\boldsymbol{C} \tag{133r}$$

$$\begin{aligned}\boldsymbol{P}_{v_{55}} = \boldsymbol{C}^T\Big\{&[(\boldsymbol{h}_1\otimes\boldsymbol{I}_3)+(\boldsymbol{I}_3\otimes\boldsymbol{h}_1)]\bar{\boldsymbol{P}}_{h_{11}}\times[(\boldsymbol{h}_1\otimes\boldsymbol{I}_3)+(\boldsymbol{I}_3\otimes\boldsymbol{h}_1)]^T\\&+[(\boldsymbol{h}_3\otimes\boldsymbol{I}_3)+(\boldsymbol{I}_3\otimes\boldsymbol{h}_3)]\bar{\boldsymbol{P}}_{h_{33}}\times[(\boldsymbol{h}_3\otimes\boldsymbol{I}_3)+(\boldsymbol{I}_3\otimes\boldsymbol{h}_3)]^T\\&-[(\boldsymbol{h}_1\otimes\boldsymbol{I}_3)+(\boldsymbol{I}_3\otimes\boldsymbol{h}_1)]\bar{\boldsymbol{P}}_{h_{13}}\times[(\boldsymbol{h}_3\otimes\boldsymbol{I}_3)+(\boldsymbol{I}_3\otimes\boldsymbol{h}_3)]^T\\&-[(\boldsymbol{h}_3\otimes\boldsymbol{I}_3)+(\boldsymbol{I}_3\otimes\boldsymbol{h}_3)]\bar{\boldsymbol{P}}_{h_{13}}^T\times[(\boldsymbol{h}_1\otimes\boldsymbol{I}_3)+(\boldsymbol{I}_3\otimes\boldsymbol{h}_1)]^T\Big\}\boldsymbol{C}\end{aligned} \tag{133s}$$

$$\begin{aligned}\boldsymbol{P}_{v_{56}} = \boldsymbol{C}^T\Big\{&[(\boldsymbol{h}_1\otimes\boldsymbol{I}_3)+(\boldsymbol{I}_3\otimes\boldsymbol{h}_1)]\bar{\boldsymbol{P}}_{h_{12}}\times[(\boldsymbol{h}_2\otimes\boldsymbol{I}_3)+(\boldsymbol{I}_3\otimes\boldsymbol{h}_2)]^T\\&-[(\boldsymbol{h}_1\otimes\boldsymbol{I}_3)+(\boldsymbol{I}_3\otimes\boldsymbol{h}_1)]\bar{\boldsymbol{P}}_{h_{13}}\times[(\boldsymbol{h}_3\otimes\boldsymbol{I}_3)+(\boldsymbol{I}_3\otimes\boldsymbol{h}_3)]^T\\&-[(\boldsymbol{h}_3\otimes\boldsymbol{I}_3)+(\boldsymbol{I}_3\otimes\boldsymbol{h}_3)]\bar{\boldsymbol{P}}_{h_{23}}^T\times[(\boldsymbol{h}_2\otimes\boldsymbol{I}_3)+(\boldsymbol{I}_3\otimes\boldsymbol{h}_2)]^T\\&+[(\boldsymbol{h}_3\otimes\boldsymbol{I}_3)+(\boldsymbol{I}_3\otimes\boldsymbol{h}_3)]\bar{\boldsymbol{P}}_{h_{33}}\times[(\boldsymbol{h}_3\otimes\boldsymbol{I}_3)+(\boldsymbol{I}_3\otimes\boldsymbol{h}_3)]^T\Big\}\boldsymbol{C}\end{aligned} \tag{133t}$$

$$\begin{aligned}\boldsymbol{P}_{v_{66}} = \boldsymbol{C}^T\Big\{&[(\boldsymbol{h}_2\otimes\boldsymbol{I}_3)+(\boldsymbol{I}_3\otimes\boldsymbol{h}_2)]\bar{\boldsymbol{P}}_{h_{22}}\times[(\boldsymbol{h}_2\otimes\boldsymbol{I}_3)+(\boldsymbol{I}_3\otimes\boldsymbol{h}_2)]^T\\&+[(\boldsymbol{h}_3\otimes\boldsymbol{I}_3)+(\boldsymbol{I}_3\otimes\boldsymbol{h}_3)]\bar{\boldsymbol{P}}_{h_{33}}\times[(\boldsymbol{h}_3\otimes\boldsymbol{I}_3)+(\boldsymbol{I}_3\otimes\boldsymbol{h}_3)]^T\\&-[(\boldsymbol{h}_2\otimes\boldsymbol{I}_3)+(\boldsymbol{I}_3\otimes\boldsymbol{h}_2)]\bar{\boldsymbol{P}}_{h_{23}}\times[(\boldsymbol{h}_3\otimes\boldsymbol{I}_3)+(\boldsymbol{I}_3\otimes\boldsymbol{h}_3)]^T\\&-[(\boldsymbol{h}_3\otimes\boldsymbol{I}_3)+(\boldsymbol{I}_3\otimes\boldsymbol{h}_3)]\bar{\boldsymbol{P}}_{h_{23}}^T\times[(\boldsymbol{h}_2\otimes\boldsymbol{I}_3)+(\boldsymbol{I}_3\otimes\boldsymbol{h}_2)]^T\Big\}\boldsymbol{C}\end{aligned} \tag{133u}$$

B. ANALYTIC EXPRESSIONS FOR ELEMENTS OF $\partial x_c/\partial b$

Given the relations from Eq. (99), it is possible to compute the partial derivatives of the calibration parameters $\boldsymbol{x}_c^T = [d_x, \alpha, d_y, u_p, v_p]$ with respect to the elements of $\boldsymbol{B}$. These partial derivatives are as follows:

$$\frac{\partial v_p}{\partial b_{11}} = -\frac{b_{23}}{b_{11}b_{22}-b_{12}^2} - b_{22}\frac{b_{12}b_{13}-b_{11}b_{23}}{(b_{11}b_{22}-b_{12}^2)^2} \tag{134a}$$

$$\frac{\partial v_p}{\partial b_{12}} = \frac{b_{13}}{b_{11}b_{22}-b_{12}^2} + 2b_{12}\frac{b_{12}b_{13}-b_{11}b_{23}}{(b_{11}b_{22}-b_{12}^2)^2} \tag{134b}$$

$$\frac{\partial v_p}{\partial b_{22}} = -b_{11}\frac{b_{12}b_{13}-b_{11}b_{23}}{(b_{11}b_{22}-b_{12}^2)^2} \tag{134c}$$

$$\frac{\partial v_p}{\partial b_{13}} = \frac{b_{12}}{b_{11}b_{22}-b_{12}^2} \tag{134d}$$

$$\frac{\partial v_p}{\partial b_{23}} = -\frac{b_{11}}{b_{11}b_{22}-b_{12}^2} \tag{134e}$$

$$\frac{\partial v_p}{\partial b_{33}} = 0 \tag{134f}$$

$$\begin{aligned}\frac{\partial \eta}{\partial b_{11}} = \frac{1}{b_{11}}\Bigg\{&\frac{1}{b_{11}}\left[b_{13}^2 + v_p\left(b_{12}b_{13}-b_{11}b_{23}\right)\right]\\&-\frac{\partial v_p}{\partial b_{11}}\left(b_{12}b_{13}-b_{11}b_{23}\right)+v_pb_{23}\Bigg\}\end{aligned} \tag{135a}$$

$$\frac{\partial \eta}{\partial b_{12}} = -\frac{1}{b_{11}}\left[v_pb_{13} + \frac{\partial v_p}{\partial b_{12}}\left(b_{12}b_{13}-b_{11}b_{23}\right)\right] \tag{135b}$$

$$\frac{\partial \eta}{\partial b_{22}} = -\frac{1}{b_{11}}\frac{\partial v_p}{\partial b_{22}}\left(b_{12}b_{13}-b_{11}b_{23}\right) \tag{135c}$$

$$\frac{\partial \eta}{\partial b_{13}} = -\frac{1}{b_{11}}\left[2b_{13} + v_pb_{12} + \frac{\partial v_p}{\partial b_{13}}\left(b_{12}b_{13}-b_{11}b_{23}\right)\right] \tag{135d}$$

$$\frac{\partial \eta}{\partial b_{23}} = \frac{1}{b_{11}}\left[v_pb_{11} - \frac{\partial v_p}{\partial b_{23}}\left(b_{12}b_{13}-b_{11}b_{23}\right)\right] \tag{135e}$$

$$\frac{\partial \eta}{\partial b_{33}} = 1 \tag{135f}$$

$$\frac{\partial d_x}{\partial b_{11}} = \frac{1}{2b_{11}}\left(\frac{\eta}{b_{11}}\right)^{-1/2}\left(\frac{\partial \eta}{\partial b_{11}} - \frac{\eta}{b_{11}}\right) \tag{136a}$$

$$\frac{\partial d_x}{\partial b_{12}} = \frac{1}{2b_{11}}\left(\frac{\eta}{b_{11}}\right)^{-1/2}\frac{\partial \eta}{\partial b_{12}} \tag{136b}$$

$$\frac{\partial d_x}{\partial b_{22}} = \frac{1}{2b_{11}}\left(\frac{\eta}{b_{11}}\right)^{-1/2}\frac{\partial \eta}{\partial b_{22}} \tag{136c}$$

$$\frac{\partial d_x}{\partial b_{13}} = \frac{1}{2b_{11}}\left(\frac{\eta}{b_{11}}\right)^{-1/2}\frac{\partial \eta}{\partial b_{13}} \tag{136d}$$

$$\frac{\partial d_x}{\partial b_{23}} = \frac{1}{2b_{11}}\left(\frac{\eta}{b_{11}}\right)^{-1/2}\frac{\partial \eta}{\partial b_{23}} \tag{136e}$$

$$\frac{\partial d_x}{\partial b_{33}} = \frac{1}{2b_{11}}\left(\frac{\eta}{b_{11}}\right)^{-1/2} \tag{136f}$$

$$\begin{aligned}\frac{\partial d_y}{\partial b_{11}} = -\frac{1}{2}\left(\frac{\eta b_{11}}{b_{11}b_{22}-b_{12}^2}\right)^{-1/2}\Bigg[&\frac{\eta b_{11}b_{22}}{(b_{11}b_{22}-b_{12}^2)^2}\\&-\frac{\frac{\partial \eta}{\partial b_{11}}b_{11}+\eta}{b_{11}b_{22}-b_{12}^2}\Bigg]\end{aligned} \tag{137a}$$

$$\begin{aligned}\frac{\partial d_y}{\partial b_{12}} = \frac{1}{2}\left(\frac{\eta b_{11}}{b_{11}b_{22}-b_{12}^2}\right)^{-1/2}\Bigg[&\frac{2\eta b_{11}b_{12}}{(b_{11}b_{22}-b_{12}^2)^2}\\&+\frac{b_{11}}{b_{11}b_{22}-b_{12}^2}\frac{\partial \eta}{\partial b_{12}}\Bigg]\end{aligned} \tag{137b}$$

$$\begin{aligned}\frac{\partial d_y}{\partial b_{22}} = -\frac{1}{2}\left(\frac{\eta b_{11}}{b_{11}b_{22}-b_{12}^2}\right)^{-1/2}\Bigg[&\frac{\eta b_{11}^2}{(b_{11}b_{22}-b_{12}^2)^2}\\&-\frac{b_{11}}{b_{11}b_{22}-b_{12}^2}\frac{\partial \eta}{\partial b_{22}}\Bigg]\end{aligned} \tag{137c}$$

$$\frac{\partial d_y}{\partial b_{13}} = \frac{1}{2}\left(\frac{\eta b_{11}}{b_{11}b_{22}-b_{12}^2}\right)^{-1/2}\frac{b_{11}}{b_{11}b_{22}-b_{12}^2}\frac{\partial \eta}{\partial b_{13}} \tag{137d}$$

$$\frac{\partial d_y}{\partial b_{23}} = \frac{1}{2}\left(\frac{\eta b_{11}}{b_{11}b_{22}-b_{12}^2}\right)^{-1/2}\frac{b_{11}}{b_{11}b_{22}-b_{12}^2}\frac{\partial \eta}{\partial b_{23}} \tag{137e}$$

$$\frac{\partial d_y}{\partial b_{33}} = \frac{1}{2}\left(\frac{\eta b_{11}}{b_{11}b_{22}-b_{12}^2}\right)^{-1/2} \frac{b_{11}}{b_{11}b_{22}-b_{12}^2}\frac{\partial \eta}{\partial b_{33}} \tag{137f}$$

$$\frac{\partial \alpha}{\partial b_{11}} = \frac{b_{12}d_x}{\eta}\left(\frac{d_x d_y}{\eta}\frac{\partial \eta}{\partial b_{11}} - 2d_x\frac{\partial d_x}{\partial b_{11}} - d_x\frac{\partial d_y}{\partial b_{11}}\right) \tag{138a}$$

$$\frac{\partial \alpha}{\partial b_{12}} = \frac{b_{12}d_x}{\eta}\left(\frac{d_x d_y}{\eta}\frac{\partial \eta}{\partial b_{12}} - 2d_x\frac{\partial d_x}{\partial b_{12}} - d_x\frac{\partial d_y}{\partial b_{12}}\right) - \frac{1}{\eta}d_x^2 d_y \tag{138b}$$

$$\frac{\partial \alpha}{\partial b_{22}} = \frac{b_{12}d_x}{\eta}\left(\frac{d_x d_y}{\eta}\frac{\partial \eta}{\partial b_{22}} - 2d_x\frac{\partial d_x}{\partial b_{22}} - d_x\frac{\partial d_y}{\partial b_{22}}\right) \tag{138c}$$

$$\frac{\partial \alpha}{\partial b_{13}} = \frac{b_{12}d_x}{\eta}\left(\frac{d_x d_y}{\eta}\frac{\partial \eta}{\partial b_{13}} - 2d_x\frac{\partial d_x}{\partial b_{13}} - d_x\frac{\partial d_y}{\partial b_{13}}\right) \tag{138d}$$

$$\frac{\partial \alpha}{\partial b_{23}} = \frac{b_{12}d_x}{\eta}\left(\frac{d_x d_y}{\eta}\frac{\partial \eta}{\partial b_{23}} - 2d_x\frac{\partial d_x}{\partial b_{23}} - d_x\frac{\partial d_y}{\partial b_{23}}\right) \tag{138e}$$

$$\frac{\partial \alpha}{\partial b_{33}} = \frac{b_{12}d_x}{\eta}\left(\frac{d_x d_y}{\eta}\frac{\partial \eta}{\partial b_{33}} - 2d_x\frac{\partial d_x}{\partial b_{33}} - d_x\frac{\partial d_y}{\partial b_{33}}\right) \tag{138f}$$

$$\frac{\partial u_p}{\partial b_{11}} = \frac{\frac{\partial \alpha}{\partial b_{11}}v_p + \frac{\partial v_p}{\partial b_{11}}\alpha}{d_y} - \frac{\alpha v_p}{d_y^2}\frac{\partial d_y}{\partial b_{11}} - 2\frac{b_{13}d_x}{\eta}\frac{\partial d_x}{\partial b_{11}} + \frac{b_{13}d_x^2}{\eta^2}\frac{\partial \eta}{\partial b_{11}} \tag{139a}$$

$$\frac{\partial u_p}{\partial b_{12}} = \frac{\frac{\partial \alpha}{\partial b_{12}}v_p + \frac{\partial v_p}{\partial b_{12}}\alpha}{d_y} - \frac{\alpha v_p}{d_y^2}\frac{\partial d_y}{\partial b_{12}} - 2\frac{b_{13}d_x}{\eta}\frac{\partial d_x}{\partial b_{12}} + \frac{b_{13}d_x^2}{\eta^2}\frac{\partial \eta}{\partial b_{12}} \tag{139b}$$

$$\frac{\partial u_p}{\partial b_{22}} = \frac{\frac{\partial \alpha}{\partial b_{22}}v_p + \frac{\partial v_p}{\partial b_{22}}\alpha}{d_y} - \frac{\alpha v_p}{d_y^2}\frac{\partial d_y}{\partial b_{22}} - 2\frac{b_{13}d_x}{\eta}\frac{\partial d_x}{\partial b_{22}} + \frac{b_{13}d_x^2}{\eta^2}\frac{\partial \eta}{\partial b_{22}} \tag{139c}$$

$$\frac{\partial u_p}{\partial b_{13}} = \frac{\frac{\partial \alpha}{\partial b_{13}}v_p + \frac{\partial v_p}{\partial b_{13}}\alpha}{d_y} - \frac{\alpha v_p}{d_y^2}\frac{\partial d_y}{\partial b_{13}} - 2\frac{b_{13}d_x}{\eta}\frac{\partial d_x}{\partial b_{13}} + \frac{b_{13}d_x^2}{\eta^2}\frac{\partial \eta}{\partial b_{13}} - \frac{d_x^2}{\eta} \tag{139d}$$

$$\frac{\partial u_p}{\partial b_{23}} = \frac{\frac{\partial \alpha}{\partial b_{23}}v_p + \frac{\partial v_p}{\partial b_{23}}\alpha}{d_y} - \frac{\alpha v_p}{d_y^2}\frac{\partial d_y}{\partial b_{23}} - 2\frac{b_{13}d_x}{\eta}\frac{\partial d_x}{\partial b_{23}} + \frac{b_{13}d_x^2}{\eta^2}\frac{\partial \eta}{\partial b_{23}} \tag{139e}$$

$$\frac{\partial u_p}{\partial b_{33}} = \frac{\frac{\partial \alpha}{\partial b_{33}}v_p + \frac{\partial v_p}{\partial b_{33}}\alpha}{d_y} - \frac{\alpha v_p}{d_y^2}\frac{\partial d_y}{\partial b_{33}} - 2\frac{b_{13}d_x}{\eta}\frac{\partial d_x}{\partial b_{33}} + \frac{b_{13}d_x^2}{\eta^2}\frac{\partial \eta}{\partial b_{33}} \tag{139f}$$

$$\frac{\partial d_x}{\partial \boldsymbol{b}} = \frac{1}{2\sqrt{\eta b_{11}}}\frac{\partial \eta}{\partial \boldsymbol{b}} - \begin{bmatrix}\sqrt{\frac{1}{\eta b_{11}^3}} & 0 & 0 & 0 & 0 & 0\end{bmatrix} \tag{140}$$

C. ANALYTIC EXPRESSIONS FOR ELEMENTS OF $\partial x_c/\partial b$

It is possible to analytically compute the partial derivatives necessary to find the covariance $\boldsymbol{P}_{gy}$ from Eq. (129). This requires computation of two derivatives, the first of which is $\partial \boldsymbol{y}/\partial \boldsymbol{h}$

$$\begin{aligned}
\boldsymbol{Z}_{yy} &\equiv \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{h}} = \begin{bmatrix}\boldsymbol{Z}_{yy_{11}} & \boldsymbol{Z}_{yy_{12}} & \boldsymbol{Z}_{yy_{13}}\\ \boldsymbol{Z}_{yy_{21}} & \boldsymbol{Z}_{yy_{22}} & \boldsymbol{Z}_{yy_{33}}\end{bmatrix}\\
\boldsymbol{Z}_{yy_{11}} &\equiv \begin{bmatrix}h_{32}u_p - h_{12} & h_{31}u_p - h_{11} & 0\\ h_{33}u_p - h_{13} & 0 & h_{31}u_p - h_{11}\\ 0 & h_{33}u_p - h_{13} & h_{32}u_p - h_{12}\end{bmatrix}\\
\boldsymbol{Z}_{yy_{12}} &\equiv \begin{bmatrix}h_{32}v_p - h_{22} & h_{31}v_p - h_{21} & 0\\ h_{22}v_p - h_{23} & 0 & h_{31}v_p - h_{21}\\ 0 & h_{33}v_p - h_{23} & h_{32}v_p - h_{22}\end{bmatrix}\\
\boldsymbol{Z}_{yy_{13}} &\equiv \begin{bmatrix}h_{12}u_p + h_{22}v_p & h_{11}u_p + h_{21}v_p & 0\\ h_{22}u_p + h_{23}v_p & 0 & h_{11}u_p + h_{21}v_p\\ 0 & h_{13}u_p + h_{23}v_p & h_{12}u_p + h_{22}v_p\end{bmatrix}\\
\boldsymbol{Z}_{yy_{21}} &\equiv 2\text{diag}([(h_{31}u_p - h_{11})\ (h_{32}u_p - h_{12})\ (h_{33}u_p - h_{13})])\\
\boldsymbol{Z}_{yy_{22}} &\equiv 2\text{diag}([(h_{31}v_p - h_{21})\ (h_{32}v_p - h_{22})\ (h_{33}v_p - h_{23})])\\
\boldsymbol{Z}_{yy_{23}} &\equiv 2\text{diag}([(h_{11}u_p + h_{21}v_p)\ (h_{12}u_p + h_{22}v_p)\\
&\qquad (h_{13}v_p + h_{23}v_p)])
\end{aligned} \tag{141a}$$

The second required derivative is $\partial \boldsymbol{g}/\partial \boldsymbol{y}$,

$$\boldsymbol{Z}_{gg} \equiv \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{h}} = \begin{bmatrix}0_{3\times 3} & 0_{3\times 3} & \boldsymbol{Z}_{gg_{13}}\\ 0_{3\times 3} & 0_{3\times 3} & \boldsymbol{Z}_{gg_{33}}\end{bmatrix}$$

$$\boldsymbol{Z}_{gg_{13}} = \begin{bmatrix}h_{32} & h_{31} & 0\\ h_{33} & 0 & h_{31}\\ 0 & h_{33} & h_{32}\end{bmatrix} \tag{142a}$$

$$\boldsymbol{Z}_{gg_{23}} = 2\begin{bmatrix}h_{31} & -h_{32} & 0\\ h_{31} & 0 & -h_{33}\\ 0 & h_{32} & -h_{33}\end{bmatrix} \tag{142b}$$

ACKNOWLEDGMENT

The authors would like to thank Harm Derksen for insightful discussions on invariant theory and Paul McKee for feedback that improved the quality of this manuscript.

REFERENCES

[1] 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.

[2] 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.

[3] D. Lang, D. W. Hogg, K. Mierle, M. Blanton, and S. Roweis, "Astrometry.NET: Blind astrometric calibration of arbitrary astronomical images," *Astronomical J.*, vol. 139, no. 5, pp. 1782–1800, May 2010, doi: 10.1088/0004-6256/139/5/1782.

[4] B. Spratling and D. Mortari, "A survey on star identification algorithms," *Algorithms*, vol. 2, no. 1, pp. 93–107, Jan. 2009, doi: 10.3390/a2010093.

[5] D. Rijlaarsdam, H. Yous, J. Byrne, D. Oddenino, G. Furano, D. Moloney, "A survey of lost-in-space star identification algorithms since 2009," *Sensors*, vol. 20, no. 9, May 2020, Art. no. #2579, doi: 10.3390/s20092579.

[6] J. L. Mundy and A. Zisserman, *Geometric Invariance in Computer Vision*. Cambridge, MA, USA: MIT Press, 1992.

[7] A. Zisserman, D. Forsyth, J. Mundy, C. Rothwell, J. Liu, and N. Pillow, "3D object recognition using invariance," *Artif. Intell.*, vol. 78, nos. 1–2, pp. 239–288, Oct. 1995, doi: 10.1016/0004-3702(95)00023-2.

[8] M. J. Ajdadi, M. Ghafarzadeh, M. Taheri, E. Mosadeq, and M. K. Ghomi, "Star identification algorithm for uncalibrated, wide FOV cameras," *Astronomical J.*, vol. 149, no. 6, p. 182, May 2015, doi: 10.1088/0004-6256/149/6/182.

[9] C. Leake, D. Arnas, and D. Mortari, "Non-dimensional star-identification," *Sensors*, vol. 20, no. 9, May 2020, Art. no. 2697, doi: 10.3390/s20092697.

[10] A. Toloei, M. Zahednamazi, R. Ghasemi, and F. Mohammadi, "A comparative analysis of star identification algorithms," *Astrophys. Space Sci.*, vol. 365, no. 4, pp. 1–9, Apr. 2020, doi: 10.1007/s10509-020-03775-9.

[11] M. D. Shuster and S. D. Oh, "Three-axis attitude determination from vector observations," *J. Guid. Control*, vol. 4, no. 1, pp. 70–77, Jan. 1981, doi: 10.2514/3.19717.

[12] F. L. Markley, "Attitude determination using vector observations and the singular value decomposition," *J. Astron. Sci.*, vol. 36, no. 3, pp. 245–258, 1988.

[13] F. L. Markley and D. Mortari, "Quaternion attitude estimation using vector observations," *J. Astron. Sci.*, vol. 48, nos. 2–3, pp. 359–380, Jun. 2000, doi: 10.1007/BF03546284.

[14] G. Wahba, "A least square estimate of satellite attitude," *SIAM Rev.*, vol. 7, no. 3, p. 409, 1965, doi: 10.1137/1007077.

[15] M. A. Samaan, D. Mortari, and J. L. Junkins, "Nondimensional star identification for uncalibrated star cameras," *J. Astron. Sci.*, vol. 54, no. 1, pp. 95–111, Mar. 2006, doi: 10.1007/BF03256478.

[16] M. A. C. Perryman, L. Lindegren, J. Kovalevsky, E. Hoeg, U. Bastian, P. Bernacca, M. Crézé, F. Donati, M. Grenon, M. Grewing, and F. Van Leeuwen, "The hipparcos catalogue," *Astron. Astrophys.*, vol. 323, no. 1, pp. L49–L52, 1997.

[17] F. van Leeuwen, "Validation of the new hipparcos reduction," *Astron. Astrophys.*, vol. 474, no. 2, pp. 653–664, Nov. 2007, doi: 10.1051/0004-6361:20078357.

[18] L. Lindegren, J. Hernández, A. Bombrun, S. Klioner, U. Bastian, M. Ramos-Lerate, A. De Torres, H. Steidelmüller, C. Stephenson, D. Hobbs, and U. Lammers, "Gaia data release 2: The astrometric solution," *Astron. Astrophys.*, vol. 616, Aug. 2018, Art. no. A2, doi: 10.1051/0004-6361/201832727.

[19] Gaia Collaboration, A. G. A. Brown, A. Vallenari, T. Prusti, and J. H. J. de Bruijne, "Gaia early data release 3. Summary of the contents and survey properties," *Astron. Astrophys.*, vol. 616, Dec. 2018, Art. no. A1, doi: 10.1051/0004-6361/201833051.

[20] 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.

[21] 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.

[22] A. Fey *et al.*, "The second international celestial reference frame by very long baseline interferometry," *Astronomical J.*, vol. 150, no. 2, 2015, Art. no. 58, doi: 10.1088/0004-6256/150/2/58.

[23] M. D. Shuster, "Stellar aberration and parallax: A tutorial," *J. Astron. Sci.*, vol. 51, no. 4, pp. 477–494, Dec. 2003, doi: 10.1007/BF03546295.

[24] J. Christian, "StarNAV: Autonomous optical navigation of a spacecraft by the relativistic perturbation of starlight," *Sensors*, vol. 19, no. 19, Sep. 2019, Art. no. 4064, doi: 10.3390/s19194064.

[25] R. Hartley and A. Zisserman, *Multiple View Geometry*, 2nd ed. Cambridge, U.K.: Cambridge Univ. Press, 2003.

[26] J. Gallier, *Geometric Methods and Applications*, 2nd ed. New York, NY, USA: Springer-Verlag, 2011.

[27] R. Zanetti, "Rotations, transformations, left quaternions, right quaternions?" *J. Astron. Sci.*, vol. 66, no. 3, pp. 361–381, Sep. 2019, doi: 10.1007/s40295-018-00151-2.

[28] 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.

[29] H. Derksen and G. Kemper, *Constructive Invariant Theory*, (Encyclopaedia of Mathematical Sciences), vol. 130, 2nd ed. Cham, Switzerland: Springer, 2015.

[30] J. A. Christian, H. Derksen, and R. Watkins, "Lunar crater identification in digital images," 2020, *arXiv:2009.01228*. [Online]. Available: http://arxiv.org/abs/2009.01228

[31] 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.

[32] P. R. Rupert, "SMART—A three-axis stabilized attitude reference technique.," *J. Spacecraft Rockets*, vol. 8, no. 12, pp. 1195–1201, Dec. 1971, doi: 10.2514/3.30364.

[33] J. L. Junkins, C. C. White, and J. D. Turner, "Star pattern recognition for real time attitude determination," *J. Astron. Sci.*, vol. 15, no. 3, pp. 251–270, 1977.

[34] C. C. Liebe, "Pattern recognition of star constellations for spacecraft applications," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 7, no. 6, pp. 34–41, Jun. 1992, doi: 10.1109/62.145117.

[35] D. Mortari, M. A. Samaan, C. Bruccoleri, and J. L. Junkins, "The pyramid star identification technique," *Navigation*, vol. 51, no. 3, pp. 171–183, Sep. 2004, doi: 10.1002/j.2161-4296.2004.tb00349.x.

[36] C. L. Cole and J. L. Crassidis, "Fast star-pattern recognition using planar triangles," *J. Guid., Control, Dyn.*, vol. 29, no. 1, pp. 64–71, Jan. 2006, doi: 10.2514/1.13314.

[37] J. Roshanian, S. Yazdani, and M. Ebrahimi, "Star identification based on Euclidean distance transform, Voronoi tessellation, and k-nearest neighbor classification," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 52, no. 6, pp. 2940–2949, Dec. 2016, doi: 10.1109/TAES.2016.150642.

[38] C. Padgett and K. Kreutz-Delgado, "A grid algorithm for autonomous star identification," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 33, no. 1, pp. 202–213, Jan. 1997, doi: 10.1109/7.570743.

[39] D. Su, D. Chen, M. Zhou, and K. Wang, "Star identification independence on the camera parameters," in *Proc. SPIE*, vol. 9284, 2014, doi: 10.1117/12.2070141.

[40] D. Forsyth, J. L. Mundy, A. Zisserman, C. Coelho, A. Heller, and C. Rothwell, "Invariant descriptors for 3D object recognition and pose," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 10, pp. 971–991, 1991, doi: 10.1109/34.99233.

[41] J. G. Semple and G. T. Kneebone, *Algebraic Projective Geometry*. Oxford, U.K.: Oxford Univ. Press, 1952.

[42] C. A. Rothwell, A. Zisserman, D. A. Forsyth, and J. L. Mundy, "Planar object recognition using projective shape representation," *Int. J. Comput. Vis.*, vol. 16, no. 1, pp. 57–99, Sep. 1995, doi: 10.1007/BF01428193.

[43] P. McIlroy, S. Izadi, and A. Fitzgibbon, "Kinectrack: 3D pose estimation using a projected dense dot pattern," *IEEE Trans. Vis. Comput. Graphics*, vol. 20, no. 6, pp. 839–851, Jun. 2014, doi: 10.1109/TVCG.2013.262.

[44] R. Lenz and P. Meer, "Point configuration invariants under simultaneous projective and permutation transformations," *Pattern Recognit.*, vol. 27, no. 11, pp. 1523–1532, Nov. 1994, doi: 10.1016/0031-3203(94)90130-9.

[45] P. Meer, R. Lenz, and S. Ramakrishna, "Efficient invariant representations," *Int. J. Comput. Vis.*, vol. 26, no. 2, pp. 137–152, 1998, doi: 10.1023/A:1007944826230.

[46] R. Hartshorne, *Algebraic Geometry*. New York, NY, USA: Springer, 1977.

[47] E. J. Groth, "A pattern-matching algorithm for two-dimensional coordinate lists," *Astronomical J.*, vol. 91, pp. 1244–1248, May 1986, doi: 10.1086/114099.

[48] D. T. Clemens and D. W. Jacobs, "Space and time bounds on indexing 3D models from 2D images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 10, pp. 1007–1017, 1991, doi: 10.1109/34.99235.

[49] G. R. Hjaltason and H. Samet, "Index-driven similarity search in metric spaces (Survey Article)," *ACM Trans. Database Syst.*, vol. 28, no. 4, pp. 517–580, Dec. 2003, doi: 10.1145/958942.958948.

[50] K. M. Górski, E. Hivon, A. J. Banday, B. D. Wandelt, F. K. Hansen, M. Reinecke, and M. Bartelmann, "HEALPix: A framework for high-resolution discretization and fast analysis of data distributed on the sphere," *Astrophysical J.*, vol. 622, no. 2, pp. 759–771, Apr. 2005, doi: 10.1086/427976.

[51] D. Arnas, M. A. A. Fialho, and D. Mortari, "Fast and robust kernel generators for star trackers," *Acta Astronautica*, vol. 134, pp. 291–302, May 2017, doi: 10.1016/j.actaastro.2017.02.016.

[52] J. L. Bentley, "Multidimensional binary search trees used for associative searching," *Commun. ACM*, vol. 18, no. 9, pp. 509–517, Sep. 1975, doi: 10.1145/361002.361007.

[53] A. Guttman, "R-trees: A dynamic index structure for spatial searching," *SIGMOD Rec.*, vol. 14, no. 2, pp. 47–57, Jun. 1984, doi: 10.1145/971697.602266.

[54] D. Arnas, C. Leake, and D. Mortari, "The n-dimensional k-vector and its application to orthogonal range searching," *Appl. Math. Comput.*, vol. 372, May 2020, Art. no. 125010, doi: 10.1016/j.amc.2019.125010.
[55] J. L. Bentley and J. H. Friedman, "Data structures for range searching," *ACM Comput. Surv.*, vol. 11, no. 4, pp. 397–409, Dec. 1979, doi: 10.1145/356789.356797.
[56] D. Mortari, "A fast on-board autonomous attitude determination system based on a new star-id technique for a wide FOV star tracker," in *Proc. AAS/AIAA Spaceflight Mech. Meeting*, no. AAS 96-158, 1996, pp. 893–904.
[57] D. Mortari and B. Neta, "K-vector range searching techniques," in *Proc. AAS/AIAA Spaceflight Mech. Meeting*, no. AAS 00-128, 2000.
[58] Y. Dong, F. Xing, and Z. You, "Brightness independent 4-star matching algorithm for lost-in-space 3-axis attitude acquisition," *Tsinghua Sci. Technol.*, vol. 11, no. 5, pp. 543–548, Oct. 2006, doi: 10.1016/S1007-0214(06)70232-2.
[59] F. Somayehee, A. A. Nikkhah, J. Roshanian, and S. Salahshoor, "Blind star identification algorithm," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 56, no. 1, pp. 547–557, Feb. 2020, doi: 10.1109/TAES.2019.2917572.
[60] J. Gerhard, "A geometric hashing technique for star pattern recognition," M.S. thesis, West Virginia Univ., Morgantown, WV, USA, 2016.
[61] J. L. Farrell, J. C. Stuelpnagel, R. H. Wessner, J. R. Velman, and J. E. Brook, "A least squares estimate of satellite attitude (Grace Wahba)," *SIAM Rev.*, vol. 8, no. 3, pp. 384–386, Jul. 1966, doi: 10.1137/1008080.
[62] J. C. Gower and G. B. Dijksterhuis, *Procrustes Problems*. Oxford, U.K.: Oxford Univ. Press, 2004.
[63] 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.
[64] 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.
[65] R. I. Hartley, "In defense of the eight-point algorithm," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 6, pp. 580–593, Jun. 1997, doi: 10.1109/34.601246.
[66] J. L. Crassidis and J. L. Junkins, *Optimal Estimation of Dynamic Systems*, 2nd ed. Boca Raton, FL, USA: CRC Press, 2012, ch. 2.
[67] 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.
[68] S. van Huffel and J. Vandewalle, "Analysis and properties of the generalized total least squares problem $AX \approx B$ when some or all columns in a are subject to error," *SIAM J. Matrix Anal. Appl.*, vol. 10, no. 3, pp. 294–315, Jul. 1989, doi: 10.1137/0610023.
[69] M. Pal and M. S. Bhat, "Autonomous star camera calibration and spacecraft attitude determination," *J. Intell. Robotic Syst.*, vol. 79, no. 2, pp. 323–343, Aug. 2015, doi: 10.1007/s10846-014-0068-z.
[70] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 11, pp. 1330–1334, 2000, doi: 10.1109/34.888718.
[71] J. R. Magnus and H. Neudecker, "The commutation matrix: Some properties and applications," *Ann. Statist.*, vol. 7, no. 2, pp. 381–394, Mar. 1979.
[72] S. Jazaeri, B. Schaffrin, and K. Snow, "On weigthed total least-squares adjustment with multiple constraints and singular dispersion matrices," *ZFV–Zeitschrift für Geodäsie, Geoinformation und Landmanagement*, vol. 139, no. 4, pp. 229–240, 2014, doi: 10.12902/zfv-0017-2014.

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. He is currently an Associate Professor with the Department of Mechanical, Aerospace, and Nuclear Engineering, Rensselaer Polytechnic Institute. He previously held an academic position at West Virginia University from 2013 to 2017 and worked as an Engineer with the GNC Autonomous Flight Systems Branch, NASA Johnson Space Center, from 2010 to 2012. He is the author or coauthor on over 100 publications. His research interests include spacecraft navigation, computer vision, and planetary science. He has contributed to many past spaceflight missions. He is a NASA Innovative Advanced Concepts (NIAC) Fellow. He is also an Associate Fellow of AIAA. He was a recipient of the AFOSR Young Investigator Award.

JOHN L. CRASSIDIS is currently a SUNY Distinguished Professor and holder of the Samuel P. Capen Chair with the Department of Mechanical and Aerospace Engineering, University at Buffalo, The State University of New York. He previously held academic positions at the Catholic University of America from 1996 to 1998 and Texas A&M University from 1998 to 2001. He also held a position as a NASA Postdoctoral Research Fellow at the Goddard Space Flight Center from 1994 to 1996. While at NASA-Goddard, he was a key contributor to the design of several spacecraft, such as TRMM, WMAP, and GOES. He is also the Director of the Center for Multisource Information Fusion, which is the only dedicated data/information fusion center in the country. He has over 250 publications and two published books. He is also a Fellow of AIAA, ASME, and AAS.