SOFTWARE FOR OPTICAL NAVIGATION AND INSTRUMENT CALIBRATION (SONIC)

Tara Mina[1], Ava Thrasher[1], Michela Mancini[1], Sébastien Henry[1], Priyal Soni[1], Jennifer Nolan[1], Benjamin Benjadol[1], and John Christian[1*] ; [1]Georgia Institute of Technology, Atlanta, GA, [*][john.a.christian@gatech.edu]

Abstract. *We introduce a new modular, object-oriented, and open-source MATLAB toolkit: the Software for Optical Navigation and Instrument Calibration (`SONIC`)[*]. With a strong basis in projective geometry and geometric algebra, SONIC provides seamless access to years of optical navigation (OPNAV) research. The toolkit's capabilities include star identification, camera and distortion modeling, geometric algebra operations and analysis, reflectance modeling, relativistic optics, state-of-the-art triangulation and pose estimation algorithms, horizon-based OPNAV, 3D reconstruction, and more. As the field of OPNAV grows and becomes increasingly interdisciplinary, we design* `SONIC` *to simplify and accelerate the production of new OPNAV research results.*

Introduction. With the rapid expansion of space exploration rendering terrestrial resources significantly constrained, there is a heightened demand for missions to integrate autonomous navigation methods into their architecture. Furthermore, autonomous navigation capabilities become critical as we venture deeper into space, where communication with Earth is subject to significant time delays. The field of optical navigation (OPNAV), with many decades of research dedicated to it, is now becoming an increasingly appealing research area to address the need for autonomy, particularly as access to improved computational ability and cameras is more widely available. By utilizing images of celestial bodies and star fields as reference points, OPNAV allows for spacecraft state estimation while reducing the reliance on Earth-based systems, such as the Deep Space Network (DSN) and other Space Communication and Navigation (SCaN) resources.

To contribute to the expanding field of OPNAV and to unify many of its principal ideas, the Space Exploration Analysis Lab (SEAL) at Georgia Tech has developed a new modular, open-source MATLAB toolkit: the Software for Optical Navigation and Instrument Calibration (`SONIC`).[1] Through the development of this toolkit, we have taken inspiration from numerous existing software libraries. Of particular note is the Goddard Image Analysis and Navigation Toolkit (GIANT),[2] which is a widely recognized Python-based OPNAV API. Utilized in the OSIRIS-REx mission, GIANT employs a scene-oriented paradigm for OPNAV, allowing users to manipulate the pose of objects within a defined scene and then execute an array of OPNAV tasks. In addition to GIANT, many individual softwares exist which address pieces of larger OPNAV problems. For example, multiple software libraries exist for geometric algebra computations, such as GABLE,[3] the Clifford Multivector Toolbox,[4] and NASA's Rigid Geometric Algebra.[5] Additionally, more general image processing tools like OpenCV,[6,7] as well as astronomical image registration software such as `tetra3` and `Astrometry.net`[8] also provide useful functionality for various OPNAV applications.

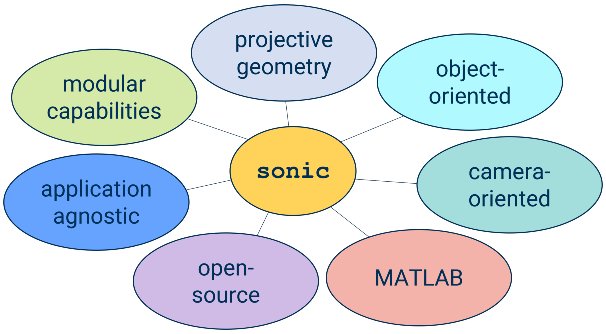

Figure 1. Overview of the design features of `SONIC`***.***

As depicted in Figure 1, we designed `SONIC` to complement these existing software tools and provide the OPNAV community with a versatile, open-sourced library that:

1. approaches the OPNAV problem from a camera-oriented perspective,
2. is grounded in principled algebraic projective geometry,[9–12]
3. integrates various state-of-the-art state estimation algorithms for OPNAV, and
4. adopts an application-agnostic, object-oriented paradigm, allowing users the flexibility to assemble OPNAV building blocks in a manner tailored to their particular research problem.

By capturing the intersection of various specialized domains (including geometric algebra, image processing, optics, and state estimation) the `SONIC` toolkit addresses the growing and increasingly interdisciplinary demands of the OPNAV research field. Our primary objective is to enable researchers to seamlessly integrate OPNAV capabilities into mission design and analysis, thus facilitating greater autonomy in future space missions.

OPNAV Functionality in `SONIC`**.** In this section, we provide an overview of the OPNAV capabilities currently supported in `SONIC`.[1] At the time of publishing, version 0.5 of `SONIC` is currently available for download on its GitHub repository.[13] Along with the base software, this download comes with multiple narrated example workflows demonstrating some applications of `SONIC` to com-

---

[*] Software available at: https://github.com/opnavlab/sonic

mon OPNAV problems. Several of these example use cases are described in this section.

Geometric Objects. Projective geometry and geometric algebra serve as the foundational elements of `SONIC`. Accordingly, the toolkit offers convenient representations of essential geometric objects in projective spaces, such as points, lines, planes, multivectors, conics, and quadrics.[9,11,12] Classical geometric computations involving these objects can be performed easily, including the analysis of their meets, joins, and projections. Additionally, `SONIC` includes several algorithms for ellipse fitting as part of its suite of conic analysis tools.[14,15]

Camera Models. The `SONIC` toolbox intuitively handles the representation of various camera models. With the `Camera` class, the user can define a camera object to process images or generate synthetic images as shown in Figure 2. In addition to the classical pinhole camera model, `SONIC` accommodates various optical distortions, such as the Brown-Conrady model.[16,17] The `Camera` class also provides flexibility in defining the camera object, allowing users to either specify the camera calibration parameters or provide the camera's field of view and image resolution.

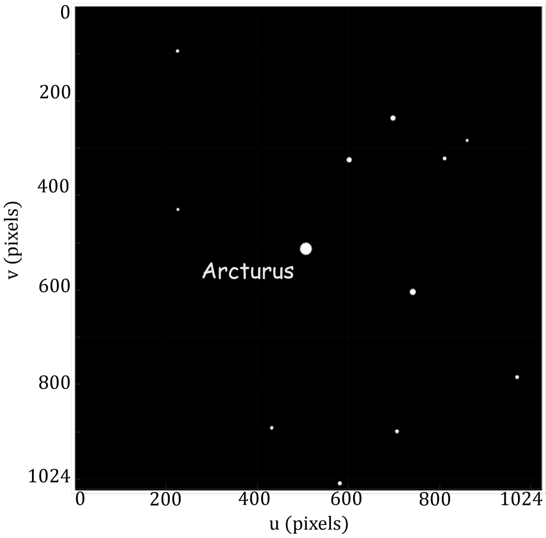

Figure 2. ***Synthetic image generated of Arcturus and the surrounding star field using a Brown-Conrady camera model and accounting for stellar aberration, with the spacecraft travelling at*** 29.8 ***km/s.***

Image Processing. The `SONIC` toolbox integrates various image processing routines specific to OPNAV, which users can easily perform in a few simple lines of code. A few notable examples are highlighted here, such as:

1. CCD smear removal,[12] as demonstrated in Figure 3.
2. Quick generation of parallel scan lines, as shown in Figure 4, or scan lines with arbitrary directions in order to analyze image intensity profiles.
3. Support for edge detection and centroiding routines, which are useful for implementations of horizon-based OPNAV, star identification, and more.

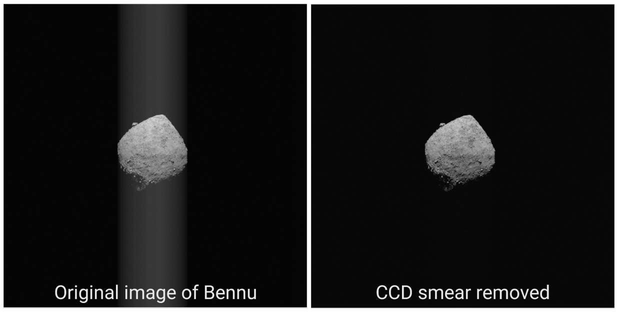

Figure 3. ***CCD image smear removal of an image of the asteroid Bennu.***

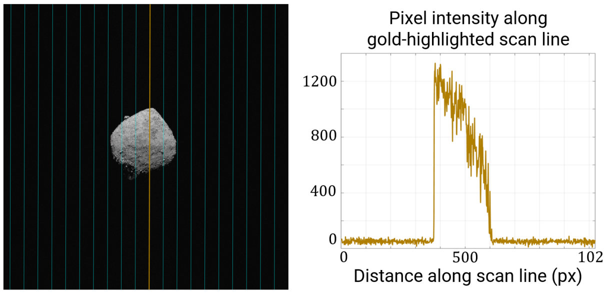

Figure 4. ***Generating parallel scan lines, and analyzing intensity profile along the gold-highlighted scan line.***

Reflectance Models. Various OPNAV techniques rely on the ability to predict how planets, moons, asteroids, and comets will appear in an image, typically requiring reflectance modeling of these bodies. Reflectance modeling has a wealth of literature supporting it, but can often become a point of confusion in implementation due to the wide range of terminology and convention used. To address this, `SONIC` provides a host of consistently implemented reflectance models to predict lighting conditions. Specifically, the current version of `SONIC` supports the Chandrasekhar,[18,19] Hapke,[20–22] Lambert,[23] Lunar-Lambert,[24,25] Lommel-Seeliger,[26] and Oren-Nayar[27] reflectance models. Figure 5 shows an example of rendering an orthographic projection of a sphere using the Lommel-Seeliger and Oren-Nayar reflectance models.

Stellar Mapping. In addition to its other capabilities, `SONIC` provides an interface to the `Hipparcos` star catalog,[28] enabling users to easily access, filter, and then evaluate the catalog. Furthermore, `SONIC` can readily incorporate stellar aberration, a relativistic optical effect, when generating a synthetic image of a star field. An illustration of synthetic image generation (accounting for both Brown-Conrady modeled distortion and stellar aberration) is shown in Figure 2, depicting the star Arcturus and its surrounding star field.

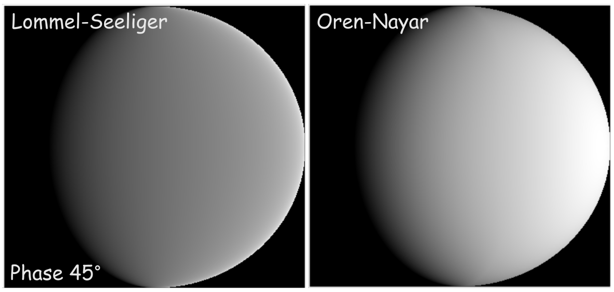

Figure 5. Rendering an orthographic projection of a sphere using the Lommel-Seeliger[26] and Oren-Nayar[27] reflectance models, at a phase angle of 45°.

State Estimation and 3D Reconstruction. Within `SONIC`, various techniques for pose estimation are readily available, offering solutions to both classical and novel OPNAV problem scenarios. These methods encompass statistically optimal triangulation algorithms,[29] horizon-based OPNAV,[30] and attitude determination from identified stars. Triangulation involves localizing the camera using known positions of key points. Conversely, by reversing this problem, one can achieve 3D reconstruction, which entails localizing key points on an object's surface using multiple images captured from known camera locations. An example of this process and the resulting 3D reconstruction is illustrated in Figure 6. In this example, we accomplished 3D reconstruction of the asteroid Vesta using the Linear Optimal Sine Triangulation (LOST) algorithm[29] and the image dataset from AstroVision.[31]

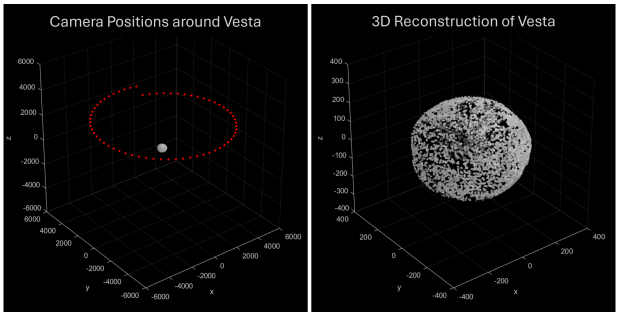

Figure 6. 3D reconstruction of the asteroid Vesta, using data from AstroVision.[31] The left plot shows the camera positions around Vesta as red points, while the right shows a close-up of the 3D reconstruction of the asteriod.

Upcoming Features. While the version 0.5 of `SONIC` is released and publicly available on GitHub,[13] this is an evolving software that will adapt to the needs of the OPNAV community and forthcoming research. In the near term, future versions of `SONIC` will include interfaces for the United States Naval Observatory (USNO) Guidance, Navigation, and Control (GNC) star catalog[32,33] and the Robbins Crater Database[34] to support crater-based OPNAV strategies.

Community Support. For those interested in contributing or reporting bugs, we invite you to visit our GitHub repository and submit an issue through the 'Issues' tab, following the provided instructions. As we strive to make `SONIC` reflect the needs of the OPNAV community, we greatly appreciate feedback to shape future releases.

References.

[1] A. C. Thrasher, M. Krause, S. Henry, M. Mancini, P. Soni, and J. A. Christian, "SONIC: Software for optical navigation and instrument calibration," *Journal of Open Source Software*, vol. 9, no. 101, p. 6916, 2024.

[2] A. Liounis, J. Swenson, J. Small, J. Lyzhoft, B. Ashman, K. Getzandanner, D. Highsmith, M. Moreau, C. Adam, P. Antreasian, *et al.*, "Independent optical navigation processing for the OSIRIS-REx mission using the Goddard image analysis and navigation tool," in *RPI Space Imaging Workshop*, no. GSFC-E-DAA-TN74562, 2019.

[3] S. Mann, L. Dorst, and T. Bouma, "The making of GABLE: A geometric algebra learning environment in MATLAB," *Geometric Algebra with Applications in Science and Engineering*, pp. 491–511, 2001.

[4] S. J. Sangwine and E. Hitzer, "Clifford multivector toolbox (for MATLAB)," *Advances in Applied Clifford Algebras*, vol. 27, no. 1, pp. 539–558, 2017.

[5] J. R. Carpenter, "Rigid geometric algebra," 2023. In GitHub repository.

[6] G. Bradski, A. Kaehler, *et al.*, "OpenCV," *Dr. Dobb's Journal of Software Tools*, vol. 3, no. 2, 2000.

[7] OpenCV, 2024. Accessed: 2024-10-03.

[8] D. Lang, D. W. Hogg, K. Mierle, M. Blanton, and S. Roweis, "Astrometry.net: Blind astrometric calibration of arbitrary astronomical images," *The Astronomical Journal*, vol. 139, no. 5, p. 1782, 2010.

[9] E. Lengyel, *Projective geometric algebra illuminated.* Terathon Software LLC, 2024.

[10] J. Richter-Gebert, *Perspectives on projective geometry: A guided tour through real and complex geometry.* Springer, 2011.

[11] J. G. Semple and G. T. Kneebone, *Algebraic Projective Geometry.* Oxford University Press, 1952.

[12] J. A. Christian, *Spacecraft Optical Navigation.* forthcoming textbook.

[13] OpNav Lab, "SONIC: Software for optical navigation and instrument calibration." `https://github.com/opnavlab/sonic`, 2024. Accessed: 2024-10-03.

[14] K. Kanatani and P. Rangarajan, "Hyper least squares fitting of circles and ellipses," *Computational Statistics & Data Analysis*, vol. 55, no. 6, pp. 2197–2208, 2011.

[15] M. Krause, J. Price, and J. A. Christian, "Analytical methods in crater rim fitting and pattern recognition," in *AAS/AIAA Astrodynamics Specialist Conference*, Space Systems Design Laboratory (SSDL), August 2023.

[16] D. Brown, "Decentering distortion of lenses," *Photogrammetric engineering*, vol. 32, no. 3, pp. 444–462, 1996.

[17] A. Conrady, "Lens-systems, decentered," *Monthly Notices of the Royal Astronomical Society, Vol. 79, p. 384-390*, vol. 79, pp. 384–390, 1919.

[18] S. Chandrasekhar, "The transfer of radiation in stellar atmospheres," *Bulletin of the American Mathematical Society*, vol. 53, no. 7, pp. 641–711, 1947.

[19] S. Chandrasekhar, *Radiative transfer*. Dover Publ, 1960.

[20] B. Hapke, "Bidirectional reflectance spectroscopy: 1. theory," *Journal of Geophysical Research: Solid Earth*, vol. 86, no. B4, pp. 3039–3054, 1981.

[21] B. Hapke, "Bidirectional reflectance spectroscopy," *Icarus*, vol. 157, no. 2, pp. 523–534, 2002.

[22] B. Hapke, *Theory of reflectance and emittance spectroscopy*. Cambridge university press, 2012.

[23] J.-H. Lambert, *Photometria sive de mensura et gradibus luminis, colorum et umbrae*. Sumptibus viduae Eberhardi Klett, typis Christophori Petri Detleffsen, 1760.

[24] A. S. McEwen, "Exogenic and endogenic albedo and color patterns on Europa," *Journal of Geophysical Research: Solid Earth*, vol. 91, no. B8, pp. 8077–8097, 1986.

[25] A. S. McEwen, "Photometric functions for photoclinometry and other applications," *Icarus*, vol. 92, no. 2, pp. 298–311, 1991.

[26] M. B. Fairbairn, "Planetary photometry: The Lommel-Seeliger law," *Journal of the Royal Astronomical Society of Canada, Vol. 99, No. 3, p. 92*, vol. 99, p. 92, 2005.

[27] M. Oren and S. K. Nayar, "Generalization of the Lambertian model and implications for machine vision," *International Journal of Computer Vision*, vol. 14, pp. 227–251, 1995.

[28] M. A. Perryman, L. Lindegren, J. Kovalevsky, E. Hoeg, U. Bastian, P. Bernacca, M. Crézé, F. Donati, M. Grenon, M. Grewing, *et al.*, "The HIPPARCOS catalogue," *Astronomy and Astrophysics, Vol. 323, p. L49-L52*, vol. 323, pp. L49–L52, 1997.

[29] S. Henry and J. A. Christian, "Absolute triangulation algorithms for space exploration," *Journal of Guidance, Control, and Dynamics*, vol. 46, no. 1, pp. 21–46, 2023.

[30] J. A. Christian, "A tutorial on horizon-based optical navigation and attitude determination with space imaging systems," *IEEE Access*, vol. 9, pp. 19819–19853, 2021.

[31] T. Driver, K. A. Skinner, M. Dor, and P. Tsiotras, "AstroVision: Towards autonomous feature detection and description for missions to small bodies using deep learning," *Acta Astronautica*, vol. 210, pp. 393–410, 2023.

[32] U.S. Naval Observatory, "International celestial reference system (ICRS)." `https://crf.usno.navy.mil/icrs`. Accessed: 2024-10-03.

[33] U.S. Naval Observatory, "GNC documentation (version 1.1)," tech. rep., U.S. Naval Observatory, 2023.

[34] S. Robbins, "A new global database of lunar impact craters $>$ 1–2 km: 1. crater locations and sizes, comparisons with published databases, and global analysis," *Journal of Geophysical Research: Planets*, vol. 124, pp. 871–892, 2019.