SSC24-P4-11

A Low-Cost, Hardware-in-the-Loop Simulator Facilitating CubeSat Star Tracker Development

Hongrui Zhao, Adrian Zhuang, Michael F. Lembeck
University of Illinois at Urbana-Champaign
hongrui5@illinois.edu; 217-250-1429

ABSTRACT

CubeSats are increasingly used to support complex missions requiring accurate attitude knowledge and pointing control. To meet these requirements, miniature star trackers are being manifested with CubeSat attitude control systems. Accurate performance verification of these trackers can be facilitated using a low-cost, hardware-in-the-loop simulator. The simulator covered in this paper incorporates a high-resolution monitor and a collimating lens to project a simulated star field for the star tracker under test. Based on a systematic accuracy analysis, this simulator is shown to be precise enough to allow for successful star identification and attitude determination in a closed loop test. An empirical method is also used to calibrate the brightness of projected stars by comparing star camera sensor outputs with night sky data. The simulator's ability to recreate operating conditions is finally validated by comparing detection and identification results against night sky data from a new star tracker now under development in the Laboratory for Advanced Space Systems at Illinois (LASSI).

1. INTRODUCTION

Star trackers match geometric patterns of star centroids on an imaging focal plane to an embedded reference star catalog to determine a spacecraft's orientation relative to the detected stars. Star trackers are also among the most accurate spacecraft attitude sensors available. Such precision is necessary for applications such as laser communications, geolocation of remotely sensed data, and object tracking [1].

A star tracker, consisting of a star camera, associated optics, and tracking algorithms, may be tested by pointing it at the night sky and comparing its attitude solutions with known star positions [2]. However, night sky testing is constrained by weather conditions and the availability of dark sky that is free from urban light pollution. The availability of a constraint free, controlled test capability can expedite the development, testing, and integration of a star tracker. Such a capability can be provided by using hardware-in-the-loop simulations that incorporate a high-definition monitor to display a simulated star field to the tracker [3].

This paper focuses on the development and verification of a low-cost simulator system incorporating commercial off-the-shelf (COTS) components for star tracker performance verification. A comprehensive analysis is presented to understand the accuracy limitations as well as display brightness considerations associated with a display screen. A validation test is also conducted to verify the capabilities of the simulator in testing the performance of a new star tracker.

Optical Simulators

Star trackers are essential instruments used in spacecraft for determining their orientation. Testing these devices is a critical step in ensuring their reliability and accuracy before they are deployed in space missions. Most methods project simulated star fields into the camera optics and focal plane.

Optical Ground Support Equipment (OGSE): Star trackers are typically tested using OGSE, which are specialized "optical stimulators" that generate synthetic images of star fields. These images are used to evaluate the star tracker's optics, electronics, and on-board attitude software. The OGSE can simulate dynamic scenarios, including the apparent motion of stars, to test the star tracker's performance under realistic conditions [3].

MINISTAR: An example of OGSE is the MINISTAR, a miniaturized device developed by a consortium of Italian enterprises and the Applied Physics Institute of the National Research Council. It's designed to test up to three star trackers simultaneously by generating synthetic images of dynamic star fields. The MINISTAR can also simulate large objects like the Sun, Earth, and Moon, as well as disturbances such as cosmic rays and stray light effects [4].

The simulator developed here similarly utilizes a commercially available display, algorithms to display

Commented [ml1]: (https://www.academia.edu/115126467/Real_time_hardware_in_the_loop_tests_of_star_tracker_algorithms?ssp=1&setlang=en&cc=US&safesearch=moderate)

Commented [ml2]: (https://www.spiedigitallibrary.org/conference-proceedings-of-spie/11180/2536202/MINISTAR--a-miniaturized-device-for-the-test-of-star/10.1117/12.2536202.full?ssp=1&setlang=en&cc=US&safesearch=moderate#_=_).

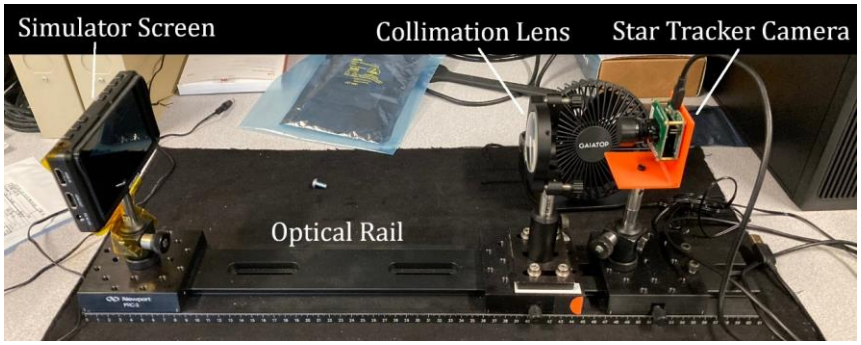

Figure 1: Star Tracker Simulator (Cover Enclosure Removed)

appropriate star fields for the tracker under test, and optics to provide displays that appear to originate at infinity. Component selection was based off the results of an accuracy analysis as well as other considerations in line with the goal of a low-cost simulator and star tracker. These star field display considerations include:

1. The angular size of the screen at the collimation distance should be larger than the field-of-view of the star tracker camera so that the full sensor can be tested.
2. The collimation lens geometry should be chosen to minimize aberrations in the projected star images to minimize errors.
3. Angular errors from discretization should be minimized.
4. Stars should be displayed at a brightness similar to their apparent magnitudes.

To meet these requirements, components were selected with the specifications provided in Tables 1, 2, and 3.

Table 1: Star Camera Specifications

Camera Sensor	CMOS Monochrome Global Shutter
Resolution	640 x 480
Focal Length	16 mm
Full Field of View	13.8 x 10.3 degrees
F/ratio	1.2

Table 2: Collimating Lens Specifications

Focal Length f_c	350 mm
Diameter	50mm
Clear Aperture	49 mm
Coating	Visual (425-675 nm)
Type	Achromatic

Table 3: Monitor Specifications

Screen	Flat 5-inch diagonal IPS
Resolution	1920 x 1080
Dot Pitch d_{pix}	0.05765 mm
Bit Depth	8
Angular Size at Collimation Distance	18 x 10.1 degrees

These components were mounted on posts, which were then placed on an optical rail, to allow for easy alignment of the system. The collimation lens is placed 350 mm away from the screen, the same as its focal length, so that the star tracker, which is focused at infinity, detects star images on the display. To prevent stray light contamination, a black enclosure covers the entire setup. A cooling fan was also included to control the test electronics' temperature inside the enclosure. As seen in Tables 1 and 3, the camera field of view matches the angular size of the screen to within 0.2 degrees. This satisfies the field-of-view consideration mentioned above. In addition, the chosen achromatic lens reduces chromatic aberration in the collimated images, fulfilling the second consideration noted above. An annotated picture of the simulator configuration is provided in Figure 1.

The developed star tracker performs star identification by using a geometric voting algorithm developed by Kolomenkin et al. [5]. In this algorithm, star pairs in the catalog vote for star pairs in the image based on similarity of angular separations. The stars in the image are assigned the identity of the stars in the catalog that cast the most votes towards them.

Based on these components and algorithms, accurately matching the near field image projected by the display to what the tracker would see in normal operations requires

an understanding of the effects of pixel discretization and magnitude calibration. Such steps are required to satisfy the third and fourth considerations.

2. ACCURACY ANALYSIS

A limiting factor for a display monitor-based star field projection is the resolution available from a digital screen. Real stars above the atmosphere appear as a point of light that is refracted and spread out as it passes through star tracker optics on the way to the focal plane. The accuracy with which a simulated star's position can be projected on the focal plane is therefore limited by the size of the pixels in the display.

To quantify the loss in positional accuracy due to limited resolution, the average angular separation between any random point within a pixel and the center of the pixel was analyzed. As a star's true location may be anywhere inside a pixel, this represents the error between the star's actual position and its projected position. An analytical approach and a computational approach were both developed to explore this problem.

2.1. Analytical Expected Error

The definition of expected value for a random variable x with a probability density function f is given by Eq. 1.

$$E[X] = \int_{-\infty}^{\infty} xf(x)dx \tag{1}$$

For a pixel centered in the optical boresight, the angle θ between the star vector to the pixel center and any point (u,v) within the pixel is given by Eq. 2.

$$\theta(u,v) = arccos\left(\frac{f_c}{\sqrt{u^2+v^2+{f_c}^2}}\right) \tag{2}$$

where f_c represents the focal length of the collimation lens. Assuming a uniformly distributed probability for a star image to be projected from within a given pixel, the probability density function remains constant over the pixel as the inverse of the pixel area. This also sets the bounds of integration to the area integral of the pixel. Combining this with (1) and (2) results in Eq. 3, which calculates the expected value of the spatial discretization error assuming square pixels.

$$E[\theta] = \frac{\int_{-0.5d_{pix}}^{0.5d_{pix}} \int_{-0.5d_{pix}}^{0.5d_{pix}} arccos\left(\frac{f_c}{\sqrt{u^2+v^2+{f_c}^2}}\right) dudv}{{d_{pix}}^2} \tag{3}$$

where d_{pix} represents the size of each pixel. This analysis calculates the error for a pixel centered with the line of sight, but this represents the worst-case scenario as perspective will make pixels appear to be smaller when they are further off-boresight, lowering the amount of discretization error by increasing perceived resolution.

Based on Eq. 3, it is seen that discretization error depends on pixel size as well as the distance to the collimation lens. Since angular accuracy is vital to star identification, the discretization error must be less than the threshold set for the geometric voting algorithm. Below this threshold, angle values are treated the same by the algorithm. It was empirically determined that a threshold of 0.03 degrees (108 arcseconds) provided reliable identifications while being flexible enough to account for other sources of error such as calibration and centroiding errors. As such, the relationship between pixel size, focal length, and expected error along with the limiting threshold can be seen in Figure 2.

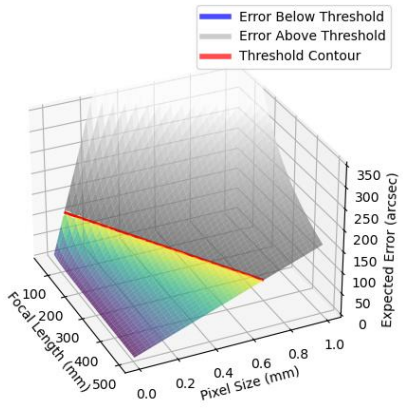

Figure 2: 3D graph showing expected error vs pixel size and collimation lens focal length. The plot is split by a contour representing the angular threshold used in the geometric voting algorithm.

As seen in the figure, higher focal lengths and smaller pixel sizes lead to less expected error. However, multiple combinations of focal length and pixel size may be used for the simulator to provide the required accuracy.

2.2. Computationally Simulated Error

To verify the results of the analytical approach as well as to understand the angular errors across the entire screen rather than just the optical boresight, a numerical approach was also developed. In this approach, all pixel positions are tracked, and random subpixel offsets are added to them to simulate potential star positions within each pixel. This was done for 250 iterations to cover a

broad range of potential offsets and resulting errors. The vector angle of each offset relative to its associated pixel center were recorded for each iteration to store the simulated angular errors at each pixel location. All iterations were then averaged, resulting in an approximation of the expected error. This approach also accounts for different resolutions, pixel sizes, and screen aspect ratios.

2.3. Error Analysis Results

Both methods described were used to evaluate the expected discretization errors of the simulator when using the selected components. Using the analytical method and Eq. 3, the expected error at the boresight evaluates to 13 arcseconds. When using the numerical method, the average expected error across the entire screen reduces to 12.89 arcseconds. A colormap of the simulated error results is shown in Figure 3.

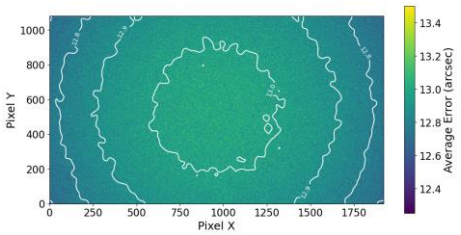

Figure 3: Colormap of Expected Screen Error. Smoothed contours highlight the decreasing error towards the edge of the screen.

Due to the nature of a flat display screen, vector angles dependent on pixel size decrease towards the edge of the display, lowering the discretization error in that area. As a result, the overall expected error across the screen is less than the analytical result. However, as anticipated by the analytical method, the center of the screen near boresight remains at 13 arcseconds of expected error in the simulation.

Based on the result of this analysis, the expected error from pixel discretization in the simulator falls well below the 108 arcsecond threshold set for star identification through geometric voting. This fulfills the third star field display consideration. As such, the effectiveness of star identification in the tested star tracker is not expected to be significantly impacted by the limited spatial resolution of the simulator screen.

3. BRIGHTNESS CALIBRATION

In order to match operating conditions, the simulator is also calibrated to display stars with the correct visual magnitude. Information about the stars themselves is stored in a file containing the Hipparcos catalog [6]. On the selected 8-bit digital screen, colors can be displayed in a brightness range from 0 to 255. The star simulator presents stars using single grayscale pixels with 255 values between black and white, with lower bit values producing fainter simulated stars. This relationship, however, needs to be calibrated specific to the setup. Previous attempts to display stars with the proper apparent magnitude used analytical calculations based on measurable quantities such as light flux from the screen [7][8]. However, these quantities may be difficult to obtain without highly sensitive measurement devices. This paper presents a more empirical method for calibrating stars displayed by the simulator. The procedure of the method, alongside a mathematical analog, is as follows:

1. Capture night sky images using the star tracker.
2. Measure raw sensor output values around star centroids in the data to determine a relationship between star magnitude and sensor output. This can be seen as:
    a. Pixel value = F(magnitude)
3. Record data of projected stars in the simulator covering the full range of brightness values to determine the relationship between display value and sensor output:
    a. Display value = G(Pixel value)
4. Combine the relationships in steps 2 and 3 to derive the relationship between display value and perceived star magnitude:
    a. Display value = G(F(magnitude))

Star magnitudes are stored on the star catalog, so this resulting relationship allows the simulator to set the brightness values of displayed stars based on their known apparent magnitudes.

3.1. Night Sky Test

To better understand how the star tracker sensor captures stars, star intensity data was recorded from a night sky test. This step is performed once to calibrate the simulator to known star magnitudes. For this demonstration, the star tracker camera was mounted on a stationary tripod aimed towards zenith and set to take 500 millisecond exposures. Data was collected at Middle Fork River Forest Preserve, IL., an area known to be largely free from urban light pollution. Star magnitude values were attained through star identifications provided by the star tracker during the test. Examples of measured stars are shown in Figure 4.

Commented [ZA3]: Hongrui suggested using mathematical analogs to help the reader understand the procedure. Let me know if this can be formatted better.

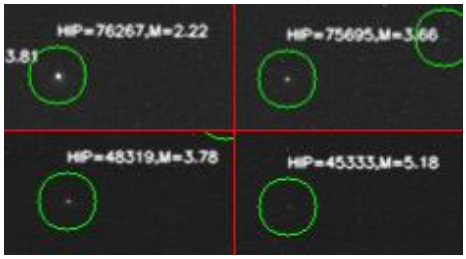

Figure 4: Cutouts of camera and star identification images from night sky data showing stars of various magnitudes. Stars with dimmer magnitudes result in darker recorded pixels by the sensor.

Dark frames were also taken with the star tracker. This involved capturing images using the same exposure settings but with the lens cap covering the aperture of the star tracker. These frames were combined into a master dark by taking the median pixel values in each frame. The master dark was subtracted from the night sky test data to reduce the effects of dark current and fixed-pattern noise. The star centroids were then measured in the resulting images. A total of 33,000 measurements were taken by finding the average pixel value in a 5x5 square around each identified star in each frame of the night sky recording. The resulting data and relationship between recorded pixel value and magnitude is shown in Figure 5.

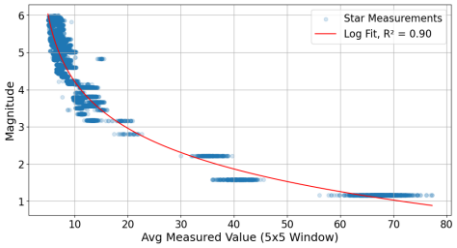

Figure 5: Average Pixel Value vs Star Magnitude from night sky data for a 500ms exposure time. Lower magnitudes produce brighter sensor values that fit a logarithmic curve well.

The horizontal patterns in the scatter plot come from measurements of the same star in multiple frames, where optical effects may alter the brightness of any individual pixel to the sensor. However, since the goal was to identify a single pixel value to use for a simulated star, a logarithmic curve fit was used to find the most appropriate relationship between magnitude and perceived brightness through the sensor output.

3.2. Screen Calibration

In the simulator, the same star tracker was used with the same exposure settings to capture images of 255 displayed stars arranged in a grid. Each star was set to a different brightness level to represent all grayscale values that were able to be shown by the display screen. Dark frames were also taken and subtracted from this data to minimize the effects of noise. Using the average pixel value in a 5x5 window around the displayed star, these results could be correlated to a respective magnitude based on the night sky data. A cutout of a raw image taken of the test grid can be seen in Figure 6.

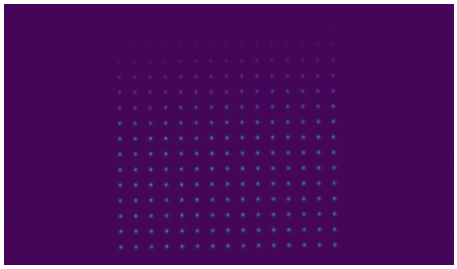

Figure 6: Star camera sensor output colormap of the pixel brightness test grid. Dimmer values are projected on the top left corner, increasing in brightness value by 1 per star towards the bottom right corner.

The 5x5 pixel windows were measured on the centroids of each projected star pixel. The relationship between the displayed pixel value and the recorded average brightness is shown in Figure 7. As seen in the figure, a clear relationship of increasing average sensor values exists for brighter displayed stars.

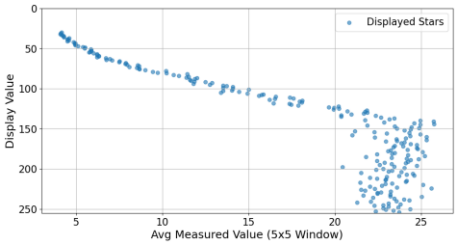

Figure 7: Displayed Pixel Value vs Average Recorded Pixel Value on sensor. To match the convention in Fig. 5, the Y axis is inverted, showing a similar relationship in brightness vs measured brightness values.

However, displayed stars brighter than value 150 appear the same to the camera. Sensor pixels start to become saturated at this point, causing the behavior to become nondeterministic. Applying this data to the logarithmic fit curve derived in Fig. 5, the final relationship is illustrated in Figure 8.

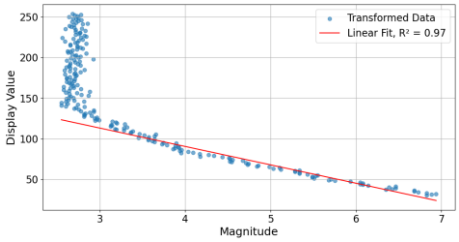

Figure 8: Displayed Pixel Value vs Estimated Pixel Magnitude. Lower pixel values are perceived as dimmer magnitude stars that follow a very linear relationship.

A line fit is generated excluding the nondeterministic stars, revealing a linear relationship that suggests the simulator can display stars as dim as magnitude seven. However, the simulator uses stars of the Hipparcos catalog limited to magnitude six, so no extrapolation of the linear fit was required. Based on these results, the simulator code was adjusted to display pixels at the values determined by this relationship.

As a result of using single pixels to represent stars and the display's limited 8-bit dynamic range, the results also indicate that brighter stars above magnitude 2.5 may be difficult to display properly. However, star identification algorithms rely more on dimmer stars when searching for star patterns in an image. As such, it is acceptable for the brightness calibration to sacrifice brighter star display fidelity in favor of displaying more accurate dimmer stars.

4. SIMULATOR VALIDATION

In lieu of an orbital test, star tracker night sky performance was compared directly to the detection and attitude computation performance obtained when viewing the same set of stars projected by the simulator. Verifying that the star tracker performs similarly to the night sky test validates that the simulator accurately projects the intended star fields.

4.1. Methods

A new star tracker under development in the LASSI lab was used to evaluate the star simulator's ability to replicate a given star field. The specifics of the new tracker's algorithm are detailed in a separate paper [9]. For the evaluation, the following metrics were used:

1. Detected Centroids – The number of stars that the star tracker can detect in each frame.
2. Number of IDs – The number of stars that the star tracker can identify in each frame.
3. Identification Rate – The percentage of the detected centroids that the star tracker can identify in each frame.

The simulator was used to display the same stars that were viewed by the star tracker in the night sky test. This ensured that the number of stars that were detectable and identifiable remained the same. Attitudes output from the star tracker algorithm were converted into quaternions that were then used to drive the simulator display. Since the data was collected on a stationary tripod, sidereal rate was induced in the displayed star field to match the motion of stars in the night sky data. Dark frames were taken and subtracted from both simulator and night sky data to reduce the effects of noise. Finally, to minimize the impact of camera lens distortion, OpenCV [10] and Matlab [11] were used to calculate the distortion coefficients of the camera and to undistort the calculated centroid positions.

Two different star fields were compared in order to determine the effects of star brightness. The first sky region was near the constellation Gemini and included stars such as Pollux and Kappa Geminorum. Since Pollux is a bright 1st magnitude star, this region was designed to test the upper extents of the brightness calibration. The second region was near the constellation Corona Borealis. The brightest star in this region was the 2nd magnitude star Alphecca, which is dimmer than Pollux. The same number of frames (141) from both regions were used in this analysis, totaling 2.35 minutes of recordings.

4.2. Validation Results

Star tracker detection and identification data was collected and stored for both the night sky and simulated sky regions. The results for the region near Gemini are seen in Figure 9, with performance averages computed in Table 4.

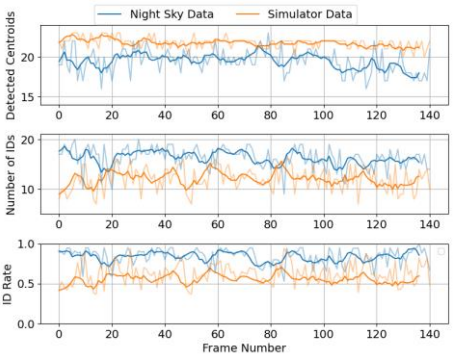

Figure 9: Detection and identification performance in the Gemini sky region for the simulator and the night sky data with a 5-frame moving average.

Table 4: Averaged Gemini Results

	Night Sky Data	Simulator Data
Avg Detected Centroids	19.45	21.75
Avg Number of IDs	16.28	12.06
Avg Identification Rate	83.78%	55.44%

In the Gemini sky region, the star tracker detects an average of 2.3 more stars from the simulator data than the night sky data. However, the identification rate of the simulator data is significantly less than the night sky data, averaging a 28% lower identification rate. For the Corona Borealis region, the results are provided in Figure 10 and the performance averaged results are computed in Table 5.

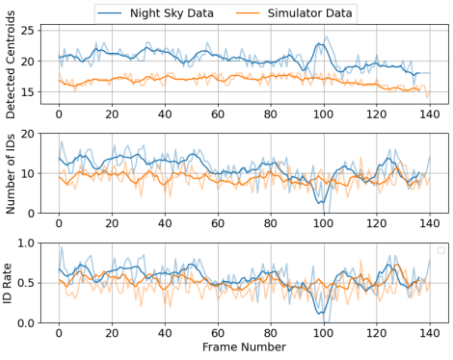

Figure 10: Detection and identification performance in the Corona Borealis sky region for the simulator and the night sky data with a 5-frame moving average.

Table 5: Averaged Corona Borealis Results

	Night Sky Data	Simulator Data
Avg Detected Centroids	20.23	16.76
Avg Number of IDs	11.25	8.70
Avg Identification Rate	56.03%	51.96%

In contrast to the Gemini region, the star tracker exhibited very similar identification rates using either the night sky or simulator data. However, the detection rates favored the night sky data for this region. The aggregated results for both regions are in Table 6.

Table 6: Aggregated Results, All Regions

	Night Sky Data	Simulator Data
Avg Detected Centroids	19.84	19.26
Avg Number of IDs	13.77	10.38
Avg Identification Rate	69.91%	53.7%

Overall, the aggregated centroid detection rates were similar; using simulator data resulted in 3% fewer detections. The night sky data produced more star identifications, leading to a 16% higher identification rate.

5. DISCUSSION

While the performance of the star tracker in each test region varied with the presentation of night sky data or simulator data, the overall results show that the detection rates for star centroids are similar. This suggests that the simulator display screen is projecting stars at appropriate brightness, resulting in similar detectability rates. While interpreting star camera sensor outputs for photometric information such as star magnitudes may not be fully accurate, it is sufficient for successful attitude determination. Of course, starlight is also attenuated by the atmosphere, reducing the accuracy of the star tracker when compared to its potential performance on orbit.

In both sky regions, the star tracker produced higher identification rates, as expected, when viewing the actual night sky data as compared to viewing the star fields generated by the simulator display. However, while the identification rates for the simulator tests remain similar for both sky regions, the night sky tests produced higher identification rates than the simulator for the Gemini region while producing similar identification rates to the simulator for the Corona Borealis region. This is likely

aided by the identification of Pollux in the star field as star centroiding accuracy increases with brighter stars [12]. Yet, the display screen is only able to accurately model the brightness of stars dimmer than magnitude 2.5, so simulated Pollux did not appear as bright as it is in the night sky. As such, the night sky performance likely benefited from increased centroiding accuracy, leading to higher identification rates. This difference between night sky Pollux and simulated Pollux can be seen in Figure 11.

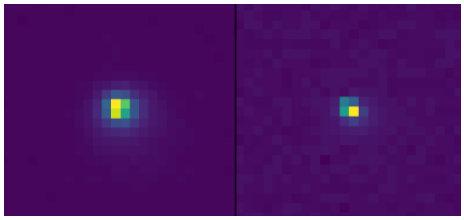

Figure 11: Raw star camera output colormap of the star Pollux from night sky data (left) and simulator data (right). Pollux is brighter and its light spreads across more pixels in the night sky data, leading to improved centroiding accuracy.

Future improvements may include a more accurate representation of brighter stars by using multiple pixels on the display screen. Additional calibration would be necessary to determine the number of pixels and their brightness values required to accurately represent 2nd magnitude stars and brighter. Still, the results in Table 5 show that identification rates for sky regions populated by dimmer stars remain similar for night sky data and for simulator data. This supports the validation of the simulator when projecting dim stars within the range of the performed brightness calibration.

6. CONCLUSION

This paper outlined a method for the construction, calibration, and verification of a star field display designed with COTS components for rapid testing and development of star trackers. The simulator was built with accuracy in mind, minimizing the amount of angular error from spatial discretization by using a high-resolution display and a collimation lens with a long focal length of 350mm. The screen was calibrated to display stars at approximately the same apparent magnitude as they would be seen on orbit. A development star tracker viewed both the simulated star field and the actual night sky to validate the accuracy of the display screen in projecting realistic star fields.

In the validation tests, the display screen was shown to project stars with a similar level of detectability to that observed in night sky testing. While not perfect, the simulator fills a gap in the star tracker developer's toolkit, providing a readily accessible and repeatable simulation of star fields for tracker functional testing. In the future, additional improvements can be made to the simulator by using photometric equipment when performing brightness calibration of the display as well as projecting especially bright stars with multiple pixels.

REFERENCES

1. Riesing, K.M. *et al.*, "On-orbit results of pointing, acquisition, and tracking for the TBIRD CubeSat mission," In *Proc. SPIE LASE*, San Francisco, CA, 2023.
2. D. Sternberg, K. Lo and J. Baker, "Night Sky Testing of the Lunar Flashlight Star Tracker," 2022 IEEE Aerospace Conference (AERO), Big Sky, MT, USA, 2022, pp. 1-13.
3. Rufino, G. *et al.*, "Real-Time Hardware-in-the-Loop Tests of Star Tracker Algorithms," International Journal of Aerospace Engineering. 2013.
4. Nardino, V. *et al.*, "MINISTAR: a miniaturized device for the test of star trackers," Proc. SPIE 11180, International Conference on Space Optics, 2018
5. M. Kolomenkin, S. Pollak, I. Shimshoni and M. Lindenbaum, "Geometric voting algorithm for star trackers," in IEEE Transactions on Aerospace and Electronic Systems, vol. 44, no. 2, pp. 441-456, April 2008.
6. Rhodes, B., "Skyfield: High precision research-grade positions for planets and Earth satellites generator", Astrophysics Source Code Library, 2019.
7. A. Reynolds, "Star Field Simulator Design." Order No. 13811874, Rensselaer Polytechnic Institute, United States -- New York, 2019.
8. Filipe, N. *et al.*, "Miniaturized Star Tracker Simulator for Closed-Loop Testing of CubeSats," Journal of Guidance, Control, and Dynamics, vol. 40, No. 12, December 2017.
9. Zhao, H. *et al.*, "Real-Time Convolutional Neural Network-Based Star Detection and Centroiding Method for CubeSat Star Tracker" arXiv preprint arXiv:2404.19108 (2024).
10. Bradski, G. "The OpenCV Library" *Dr. Dobb's Journal of Software Tools*, 2000.
11. The MathWorks, Inc. "Remove Distortion from an Image Using Camera Parameters Object" mathworks.com. [Online].

12. C. C. Liebe, "Accuracy performance of star trackers - a tutorial," in IEEE Transactions on Aerospace and Electronic Systems, vol. 38, no. 2, pp. 587-599, April 2002.