LOST: Open-Source Star Tracker

Karen Haining, Kenneth Yang, Edward Zhang
Advised by Prof. Kristi Morgansen and Mark Polyakov

LOST Background

LOST is a project created by the University of Washington's Husky Satellite Lab.

About the Husky Satellite Lab

Our mission is to foster interdisciplinary student participation in space systems research, to inspire and train future space scientists and engineers, and to advance spacecraft capabilities at UW. We are composed primarily of UW undergraduate and graduate students, as well as mentors from the local aerospace industry. Our current objective is to build a 3U CubeSat—essentially a miniature satellite—intended for deployment in low earth orbit around 2024.

Abstract

The central goal of our project is to develop an open-source star tracker which can be used by any engineering lab to solve for a CubeSat's attitude, i.e., determining its orientation. A star tracker is a combination of hardware and software that takes a photograph of the stars and uses constellations in the image to determine which direction the camera was pointing when the photograph was taken, thereby determining the attitude of the CubeSat. The star tracking process can be divided into three main stages: centroid detection, star identification, and attitude estimation. We have implemented various algorithms to accomplish each stage. Each algorithm's runtime, accuracy, and memory usage can be measured, and tradeoffs between the three performance metrics can be weighed to determine an optimal combination of algorithms.

Why is a Star Tracker Important?

Determining orientation is a critical feature for a CubeSat. Star trackers in particular are well-suited for this task, being much more accurate than other methods of attitude estimation. For example, a star tracker can continue functioning when several stars are out of view, and even when far away from any celestial body, unlike a magnetometer or sun sensor. Moreover, star trackers do not require special hardware to operate—they simply work off of an image that can be captured with a satellite camera.

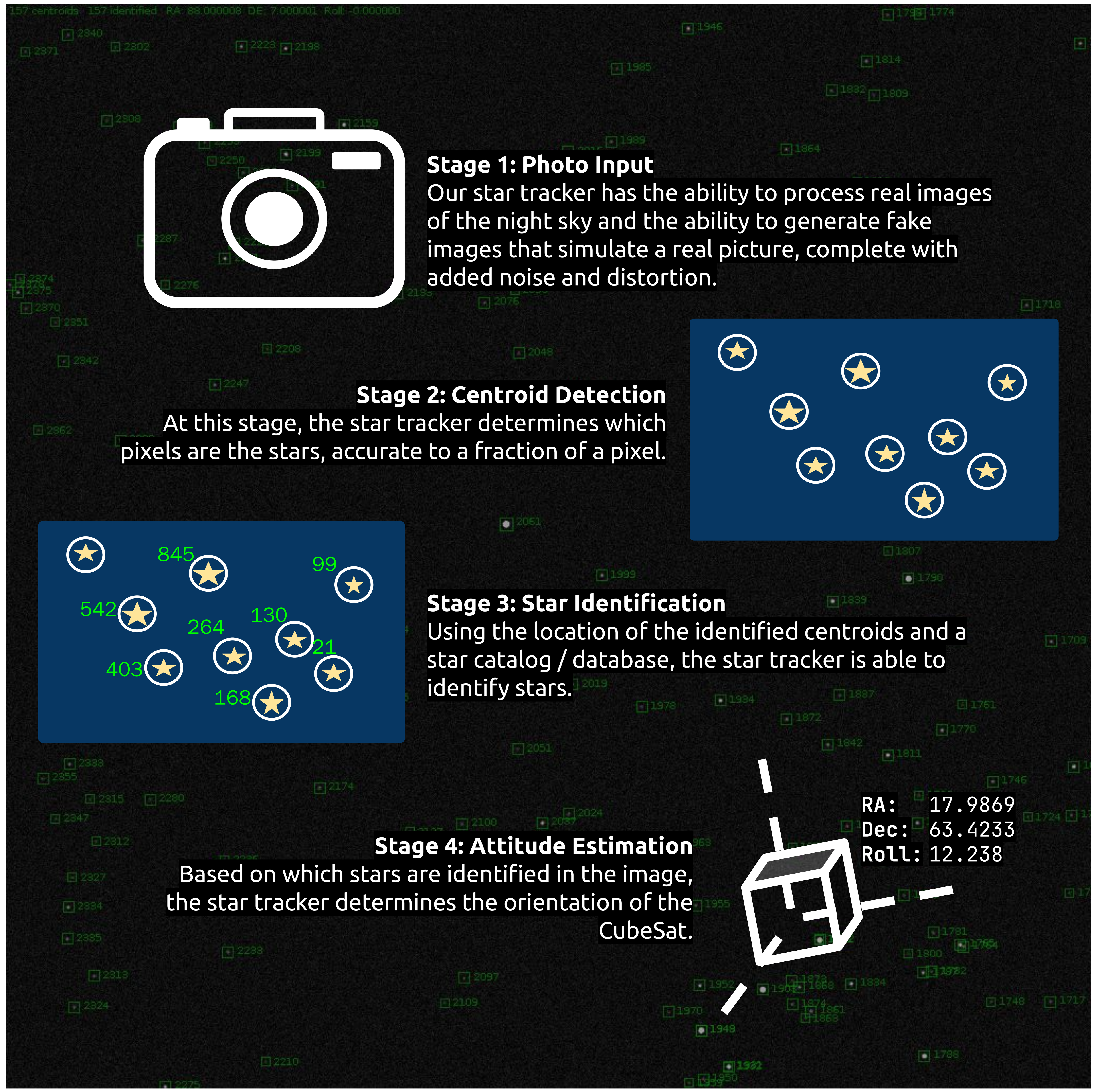
Acknowledgements

We would like to thank Washington NASA Space Grant Consortium for funding, as well as our larger team at the Husky Satellite Lab.

References

The following sources were invaluable to our research:

- Brown, Julian and Keaton Stubis. "TETRA: Star Identification with Hash Tables." *31st Annual AIAA/USU Conference on Small Satellites, Utah State University, 2017.* https://digitalcommons.usu.edu/cgi/viewcontent.cgi?article=3655&context=smallsat.
- Cheng, Yang and Malcolm D. Shuster. "Improvement to the Implementation of the QUEST Algorithm." *Journal of Guidance, Control, and Dynamis*, vol. 37, no.1, 2014, pp. 301–305.



SURP Accomplishments

The following are several major improvements and additions we made to LOST's code base over the course of SURP.

Documentation and Automation

As open-source software, LOST must be easily accessible and ready for public contribution. To enable this, we added automated linting and tests through GitHub Actions, as well as extensive documentation with Doxygen, to the project's publicly available GitHub repository. On our end, this allows us to automatically vet contributions for robustness and style. For outside contributors, documentation provides guidance in navigating the large code base, allowing developers to have an easier time understanding what classes and functions they need to achieve their goals.

Star Identification

We implemented Tetra, a new star identification algorithm whose runtime is asymptotically optimal. Tetra constructs 4-star "patterns" and stores them in a hash table, ensuring fast star matching with minimal database accesses. In addition, Tetra is also more accurate and resistant to common sources of error, such as FOV error and double stars. The tradeoff, however, is that Tetra's database is significantly larger than that of any other star ID algorithm.

In addition to Tetra, we also implemented a "tracking mode" star ID algorithm. As opposed to the "lost in space" algorithms that LOST previously used, the new algorithm allows us to determine attitude based on not just the current image, but also using information about the CubeSat's previous attitude.

Attitude Estimation Algorithms

We implemented the QUEST attitude estimation algorithm, which improves upon our previously-implemented Davenport Q method in terms of speed. Moreover, it does not require the use of external libraries to solve complex equations, which is better for a low-memory and low-power device such as a CubeSat. QUEST also carries historical importance as the de facto standard algorithm.

Next Steps

Although SURP draws to a close, we will continue introducing new features to LOST. Our plans include further optimizing the code for embedded systems and space deployment, as well as adding unit tests and performance benchmarks in order to quantify how well our software works.

Get in Touch!

Scan the left QR code to access the LOST repo at https://github.com/UWCubeSat/lost and the right QR code to access HSL's website at https://huskysat.org/!