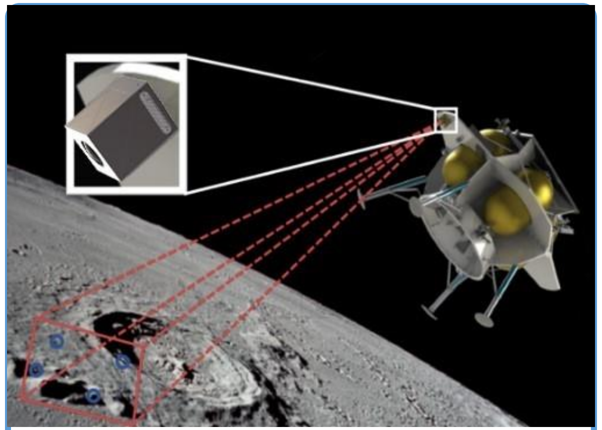
Ultra-Low SWaP Relative Navigation

Chris Owens (chris.owens@astrobotic.com), Morgan Redfield, Michael Bloom, Andrew Horchler

Astrobotic



INTRODUCTION

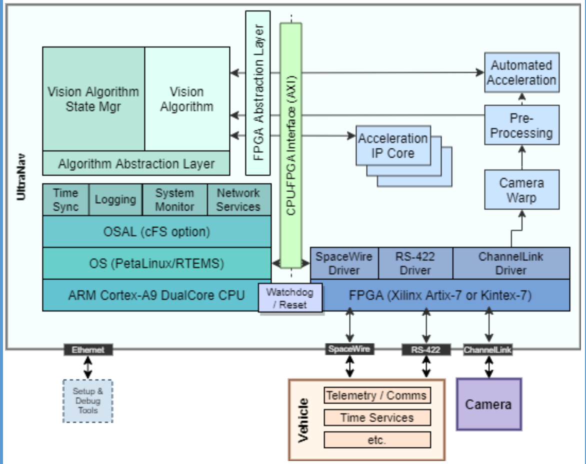
UltraNav is a low size, weight, power, and cost (SWaP-C) compute-enabled camera system capable of implementing industry standard computer-vision algorithms such as object recognition and tracking, state-estimation, and relative navigation. This system simplifies advanced missions that require capabilities such as Terrain Relative Navigation (TRN), visual Simultaneous Localization and Mapping (SLAM), 3D reconstruction, or fiducial targeting. Algorithms such as these have been limited to large spacecraft on high budget missions due to processing requirements and the resulting increase in SWaP-C. The sensor's flexibility allows it to be used for low level image capture, image processing, or as a host for a full GN&C solution.

Current large-scale missions are demonstrating the importance of high-performance vision-based algorithms for accomplishing cutting edge, high priority science objectives. OSIRIS-REx and Hayabusa2 have performed precise operations at previously unexplored asteroids, and Mars Perseverance is expected to be the most precise automated landing on a non-terrestrial planetary body to date. Vision applications such as TRN and 3D modeling will open up higher-level autonomy capabilities for small satellites missions that have not had these available to date. Prior visual navigation tools available to small spacecraft have tended to be application specific (e.g., star trackers) or have required extensive engineering effort (e.g., mission-specific ad hoc integration and algorithm development). This limited the operation of small spacecraft in GPS-uncertain environments. UltraNav promises high capability while maintaining low cost, and ease-of-integration. Additionally, UltraNav's reconfigurability enables it to be used for multiple applications in the same mission, such as switching from localization to star tracking.

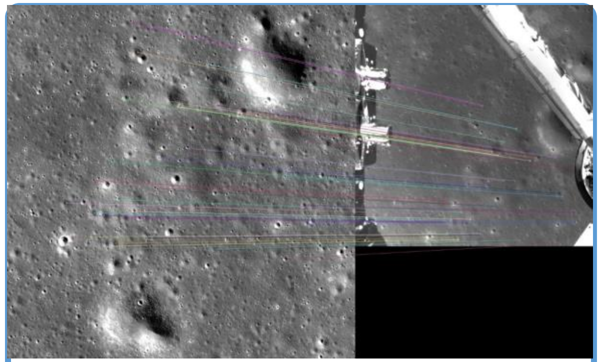
SYSTEM DESCRIPTION

The UltraNav hardware has a compact form factor fitting within 1.5U (10 × 10 × 15 cm), weighs less than 2 kg, and requires less than 5 W to power, enabling its use in resource constrained CubeSats and SmallSats. UltraNav integrates a radiation-tolerant and space-heritage dual core CPU and FPGA with an IMU and a navigation-grade camera. Vision algorithms are designed to run on both CPU and FPGA hardware, allowing for power- and compute-efficient operation that maintains operational flexibility. UltraNav provides mission designers with highly optimized vision algorithms, while allowing for the use of custom mission-specific software. UltraNav software leverages heritage flight code, such as NASA's Core Flight System (cFS), for application development, which ensures reliability and simplifies interoperability with the spacecraft.

The primary goal of UltraNav's software design is to produce a system that simplifies implementation of a variety of vision-based algorithms and integration with other devices. To accomplish this, UltraNav provides a clean and simple application program interface (API) to algorithm designers. UltraNav is provided with a set of software build-pipelines and support libraries that customers can use without extensive knowledge of the hardware. UltraNav can use PetaLinux (a Yocto derivative) or RTEMS as its operating system. While PetaLinux is not a real time operating system (RTOS), it will be suitable for many algorithms that do not have hard real time constraints. When those constraints do exist, RTEMS can be used instead, aiming for flexibility and low complexity.



The UltraNav platform will separate vision algorithm functionality between the processor subsystem and the FPGA subsystem. Much of the infrastructure chosen to support this (e.g., PetaLinux, AXI interfaces) is open source, decreasing cost and development times.



Visualization of feature extraction and matching component of terrain relative navigation on lunar imagery. The image on the right shows a subset of the lunar surface from the left image. Matched features between the images are shown connected by colored lines. Testing has also included datasets from previous sensor demonstration flights on a Masten Xombie vertical takeoff vertical landing vehicle.

VISION ALGORITHM SUPPORT

A set of abstraction applications run atop the operating system. These applications handle message passing, time keeping, logging, etc. This enables algorithm developers and mission designers to design for UltraNav without concern for the underlying operating system or low-level hardware management routines. NASA's Core Flight System (cFS) software is the primary flight software framework supported by UltraNav. Other flight software frameworks will be supported on an as-needed basis.

The motivation is to allow algorithm developers to write code once, run unit-tests and simulations on a local machine, and cross-compile the same code (excluding hardware accelerations) for the UltraNav hardware itself. PetaLinux supports this workflow for Linux applications by inheriting features from Yocto that allow target git repositories to be integrated into the OS image during the build process. Thus, any code that is released to a target repository can be automatically integrated into the next PetaLinux build. This minimizes the introduction of errors between algorithm development and implementation.

UltraNav includes several hardware design language (HDL) stubs that are placed on the FPGA fabric. These stubs allow for automated hardware acceleration tools (i.e., Xilinx's Vitis and SDSoC software packages) to be used to generate FPGA IP cores to speed up vision-based algorithms. Standard interfaces are used when manually created acceleration algorithms are used.

APPLICATIONS

Small spacecraft that integrate UltraNav will gain a flexible and cost-effective compute-enabled camera, as well as a suite of hardware-accelerated computer vision algorithms optimized for common space missions.

The initial application for UltraNav is Terrain Relative Navigation (TRN). In this navigation scheme, captured images are fed to the FPGA for image warp and distortion correction to reduce the effects of the camera optics and variations in lighting conditions. TRN software then identifies regions of interest that are processed into signatures. These signatures, or "fingerprinted" terrain landmarks, are matched against a database of scale- and rotation-invariant descriptors in a stored map to estimate the absolute position and orientation of the spacecraft relative to the planetary body. These signatures can determine matches even in the presence of large altitude, attitude, and illumination errors and thus are applicable over large envelopes.

Other applications are also under development across a variety of domains, supporting capabilities such as Rendezvous and Proximity Operations (RPO) and deep-space navigation. Current work also emphasizes generic spacecraft relative state estimation techniques.

The UltraNav platform is composed of a camera (with optics) and a compute element that runs machine vision algorithms. Together, this hardware takes up a volume of approximately 1.5U. Two different enclosure options are under investigation.

ACKNOWLEDGEMENTS

Astrobotic acknowledges support from NASA via SBIR 80NSSC19C288.