Design of an In-Situ Sensor Package to Track CubeSat Deployments

Aaron P. Aboaf, Nicholas W. Renninger, Lara C. Lufkin
University of Colorado Boulder
1111 Engineering Drive, Boulder, CO 80309
aaron.aboaf@colorado.edu

Faculty Advisor: Prof. Penina Axelrad
Professor, University of Colorado Boulder

ABSTRACT

Currently, radar measurements of low-earth-orbit CubeSats are only possible during a small portion of a CubeSat's orbit – typically long after the CubeSat's deployment – making near real-time space situational awareness (SSA) difficult. The CU Boulder Smead Aerospace Engineering Department has developed a concept to monitor CubeSat deployments from the deployer itself and provide relative position and velocity measurements of deployed payloads to provide faster orbital parameter estimation. Teaming with NanoRacks LLC, the VANTAGE team (Visual Approximation of Nanosat Trajectories to Augment Ground-based Estimation) has developed an innovative sensor package prototype consisting of an Infra-red (IR) Time of Flight (ToF) camera for close-range CubeSat position measurements and a monochrome optical camera for continued detection and in-plane position refinement, as well as a set of algorithms to process and fuse these CubeSat position measurements. These sensors and their avionics are incorporated into a prototype integrated system designed to fit within a single 6U CubeSat Deployment silo on the NanoRacks ISS deployer, enabling the detection, identification, and tracking of up to 6 CubeSats out to 100m with a maximum positional error of 10m within 15 minutes of deployment.

INTRODUCTION

As space becomes a more popular location to explore and utilize, the number of spacecraft in orbit around Earth continues to climb. As these numbers climb, building a reliable and timely awareness of the these objects and their trajectories, termed Space Situational Awareness (SSA), is critical.

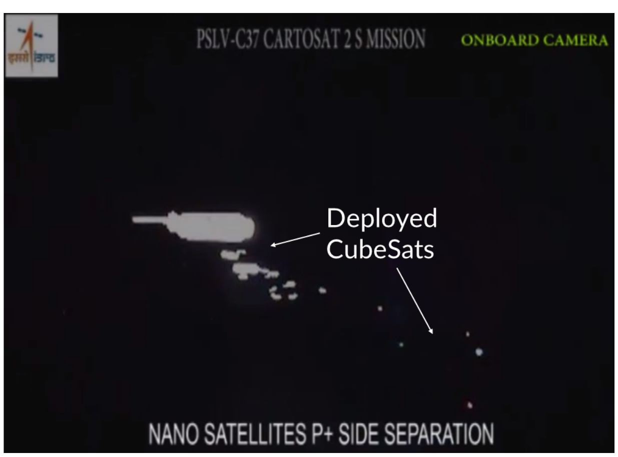

Figure 1: Image of an Indian Space Agency CubeSat Launch

One significant contribution to the increasing number of objects in orbit is the launch of CubeSats. These small satellites are sometimes launched into orbit hundreds at a time. Figure 1 shows a still image from a video taken of the Indian Space Agency CubeSat launch of 104 CubeSats [1]. As seen in this photo, the CubeSats are extremely numerous and difficult to see as they reach further distances. Ground based tracking systems are used to gather information about the trajectory of these CubeSats as they orbit. These tracking systems consist primarily of telescopes scattered across the globe. Due to the small size of these CubeSats, more sophisticated telescopes are necessary to track their orbits. These telescopes have an extremely high demand that makes it difficult to reserve viewing time. The complex process of ground based tracking can require days or more before trajectory data is gathered for each CubeSat [2]. If launch does not go as expected, the CubeSat may have an off-nominal trajectory and can no longer be tracked by ground stations. This problem leads to a significant number of CubeSats with unknown locations leading to degraded SSA.

At the Colorado Center for Astrodynamics Research (CCAR) at the University of Colorado Boulder, researchers are investigating alternatives to current CubeSat tracking methods [3][4]. To further develop this work, the VANTAGE Senior Projects team is being sponsored to build a tracking system that uses cameras to gather trajectory information from the perspective of the CubeSat deployment system. This tracking system is designed to interface with NanoRacks ISS CubeSat Deployer (NRCSD) as a future platform to demonstrate the capabilities of the tracking system design. Access to the the CubeSats' trajectory information shortly after launch increases SSA by dramatically decreasing the time it would normally take to gather information about the CubeSat's orbits. The long term vision of the VANTAGE project is to augment existing, ground-based tracking systems by observing CubeSat deployments from the perspective of the space based deployment system. This year the VANTAGE team is taking a first step towards this realization by producing a ground-based proof of concept which is tested using simulated CubeSat launches.

DESIGN OBJECTIVES

This ground-based proof of concept design is driven by several high level project objectives: i) identify, image, and track up to 6 CubeSats which are not obfuscated released at 1-2m/s out to a range of 100 m; ii) record all necessary data, process it, and return measurements within 15 minutes of receiving a start command; iii) mechanically integrate and fit within the space provided by one NanoRacks ISS CubeSat Deployer. In order to accomplish these project objectives, the design is subject to derived design objectives. While there are many design objectives which drive a final design, the specific design objectives shown in Table 1 capture the novel technical aspects of the project upon which success is most pivotal.

These specific design objectives are dependent upon the accuracy limits imposed by the project requirements. There is a requirement on the absolute error of the relative position and velocity measurements as shown in Fig. 2. This accuracy requirement applies to the final reported values of the VANTAGE system. Thus, the combined error of the sensor system and the software methods must lie below these requirements. A sensor system must be selected which complies with the project objectives and also has a high enough accuracy to accommodate the software system errors.

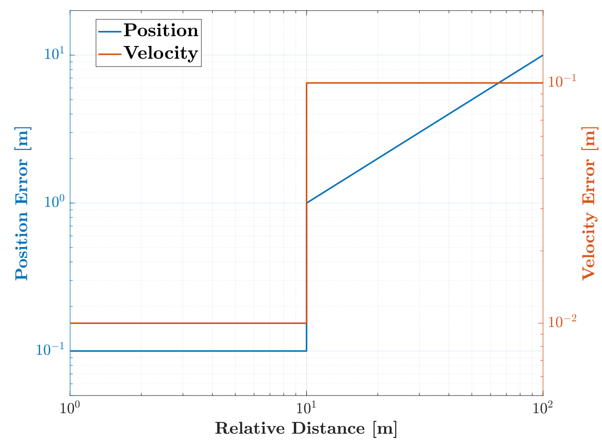

Figure 2: CubeSat Centroid Position and Velocity Error Requirements

Table 1: Specific Design Objectives

Design Objective	System	Objective Description
Sensor Accuracy Limits	Hardware	The physical hardware must be capable of providing measurements of the CubeSats within the accuracy limits imposed by the project requirements.
Object Identification	Software	Using the data provided by the physical sensors, the software system must be able to correctly identify CubeSats.
Multi-Object Tracking	Software	Having correctly identified all CubeSats in frame, the software system must be able to properly track individual CubeSats between measurements.

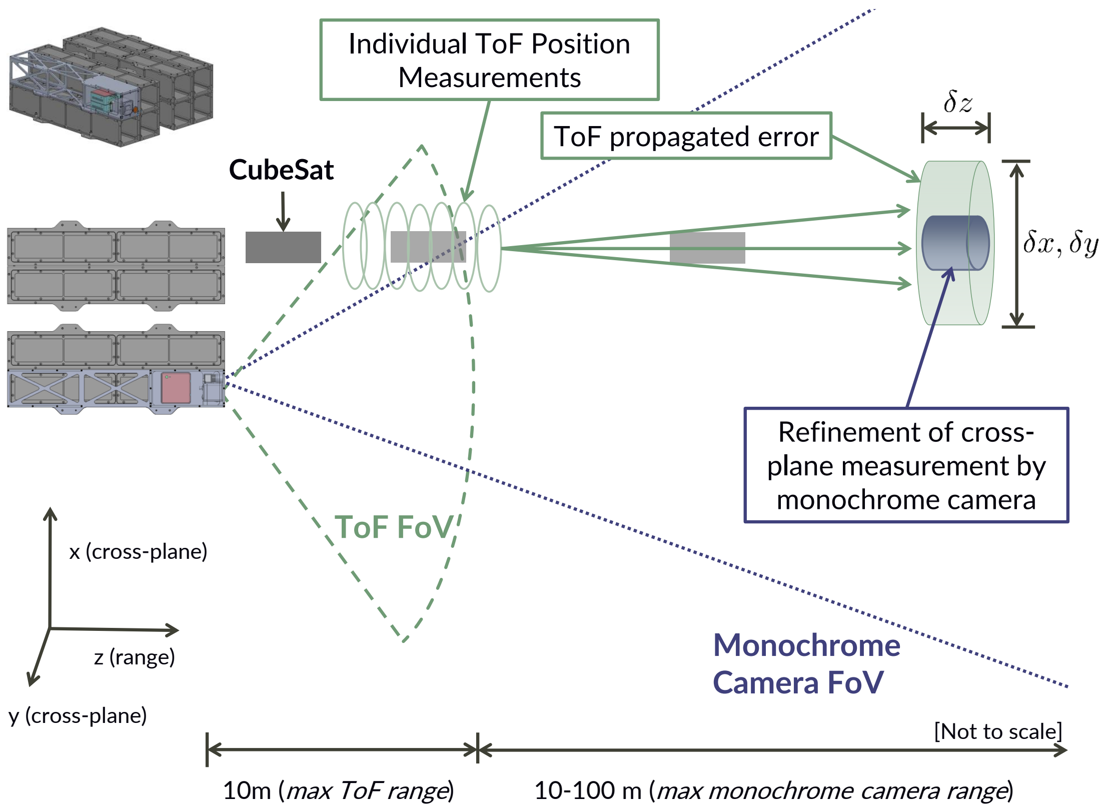

Figure 3: Operational Concept of the VANTAGE Combined Sensor Design

Centroid Extraction	Software	The software system must be capable of determining each CubeSat's centroid within the accuracy limits imposed by the project requirements.

DESIGN CONCEPT

Proposed Use-Case

The diagram shown in Fig. 4 is an illustration of the long term VANTAGE mission. To demonstrate a practical proof of concept, the NanoRacks deployer is used as a prototypical platform which constrains hardware, power, and data transfer requirements. The NanoRacks CubeSat deployment system consists of up to eight deployment tubes in total [5]. The VANTAGE system is designed to fill the space of one deployment tube, providing the customer with initial CubeSat trajectory information more quickly after launch.

The high level VANTAGE baseline design consists of three main components, a structural interface with the NanoRacks deployer, an avionics/software system, and a data acquisition system. The structural interface is designed out of quarter inch aluminum plates that integrate with the NanoRacks deployer in place of a deployment tube. The avionics/software system consists of Intel's Next Unit of Computing (NUC) which functions as the main processor for the VANTAGE system. The high-level coding language, MATLAB, is used as the main software package on this processor. Finally, a Time of Flight (ToF) and Monochrome Camera make up the data acquisition system for VANTAGE. The ToF Camera uses infrared light to acquire depth information similarly to how a sonar system uses sound to detect depth. Figure 4 provides more detail about the high-level design of the VANTAGE system.

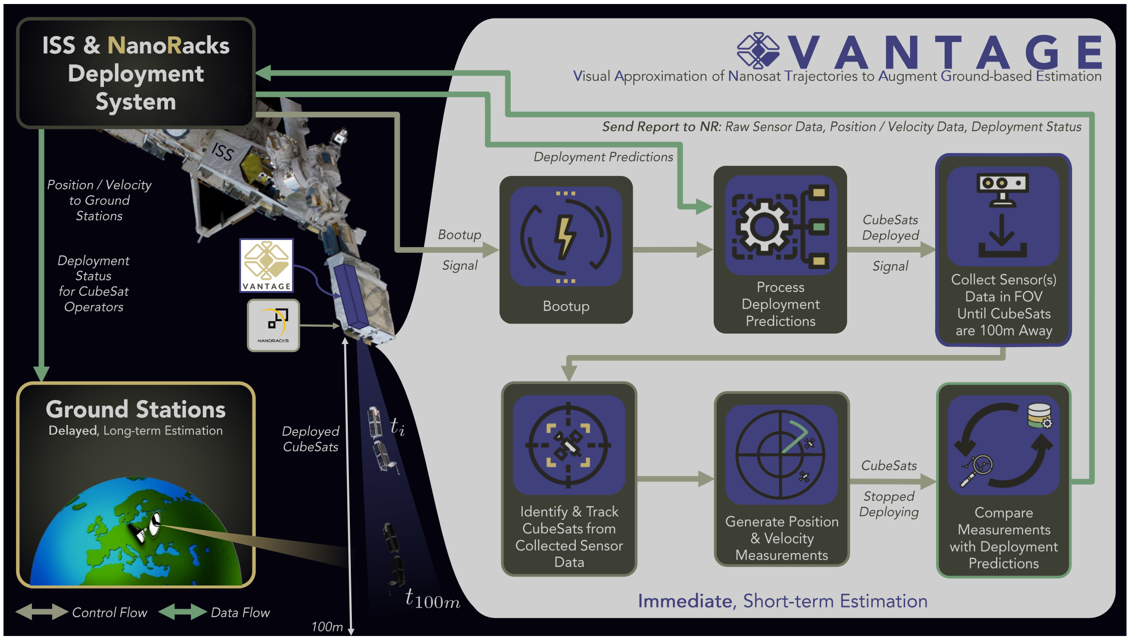

Figure 4: VANTAGE Mission Overview

Sensor Solution

The Functional Block Diagram in Fig. 6 shows the conceptual operation of the VANTAGE system integrated structure, software, and sensors. The Sensor Suite, containing the ToF and Monochrome Cameras, is designed to observe CubeSats out to 100m past the launch point within a 20° field of view. Raw images from both cameras are sent to the Command and Data Handling Suite. Here, the CubeSats are identified and measurements of their relative positions and velocities are calculated and stored onboard VANTAGE. The VANTAGE system is required to process all raw images and report these relative velocities and positions back to a mock NanoRacks deployment system within 15 minutes after the last CubeSat launch. Additional details of VANTAGE are shown in Fig. 6.

The accuracy requirements at 100m drive the need for both high accuracy at small ranges and detection capability at long ranges. In order to achieve the desired accuracy, two fundamentally different sensors are employed. The first, a ToF Camera, supports close-in ranging accuracy extending out to roughly 10m. These measurements baseline the software solution and position propagation algorithms. The second, a Monochrome Camera, provides cross track refinement of the objects' motions as they move down the range. This combination of a ToF Camera and Monochrome Camera, as seen configured in the designed system in Fig. 5, provides sufficient sensory input to the software solution. The sensing concept is laid out in Fig. 3 which describes the relevancy of each sensor measurement to the overall software solution.

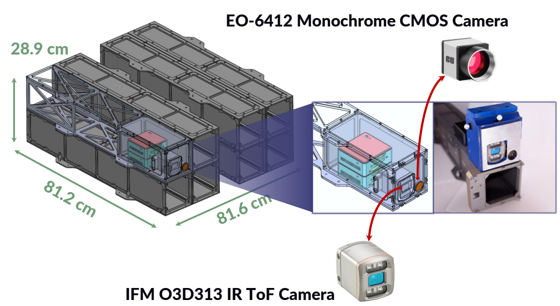

Figure 5: Physical Design and Placement of Sensors in the VANTAGE Structure.

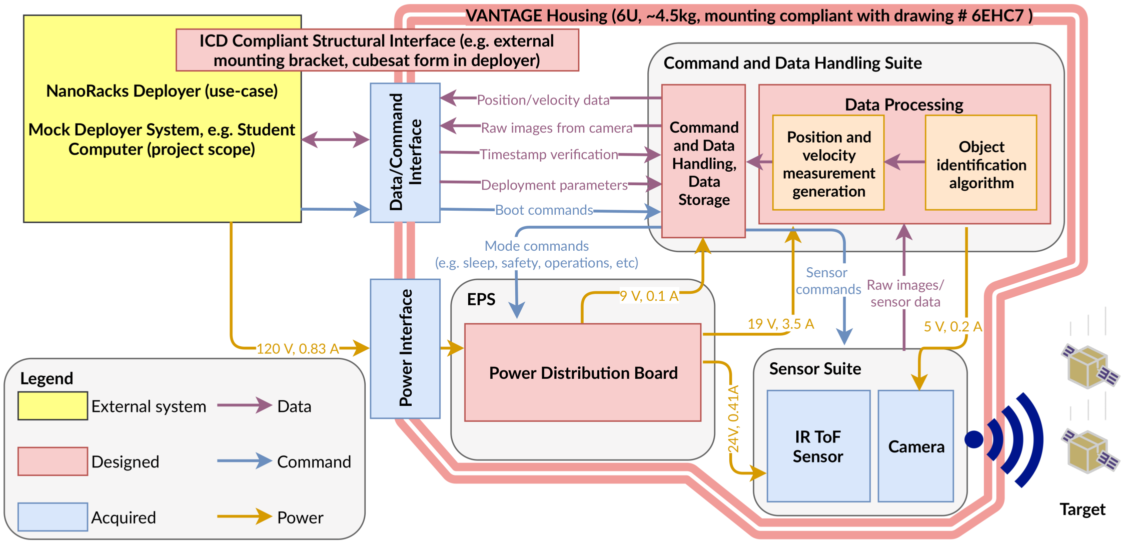

Figure 6: VANTAGE Functional Block Diagram

CUBESAT STATE CALCULATION

The VANTAGE software's most important function is to produce centroid estimates for individually identified CubeSats during a deployment. The architecture is set up for autonomous data collection of a deployment given a deployment manifest and initiation signal. This data collection from both sensors continues until the CubeSats have reached 100m. Once data collection is complete, post-processing begins, and the data from both sensors is fused together to estimate the relative position and velocity of the deployed CubeSats.

Figure 7 shows the UML design of the post-processing software, and where the novel algorithms discussed in the coming sections fit into the overall post-processing flow. The software starts by finding the 3D centroids of all individual CubeSats in the point cloud data, then does initial image processing of the monochrome images. Following this, using the 3D centroids produced from the ToF and the known dynamics of the CubeSats, predictions of the states of the CubeSats are propagated out to the full 100m range requirement. Then, the monochrome images are processed and visual (2D) centroids are extracted and fused with these 3D state estimates to create a final 3D deterministic state estimate for every monochrome image.

ToF Algorithms

To calculate centroid values the from the point clouds, the downrange point density is determined and the local minima are identified. Then the point clouds are split into distinct CubeSats, where each subset of point clouds are fit with up to three orthogonal planes corresponding to distinct faces of a CubeSat. The planes are then used to find the tightest fitting bounding box, which is used to find the centroid per face. Using these centroids, the centroid location of each CubeSat can be calculated. The pseudocode for these steps is given in Algorithm 1, with the corresponding results shown in Fig. 9.

Monochrome Camera Algorithms

As shown in Fig. 7, the monochrome camera performs processing of individual frames collected during testing. CubeSats will be searched for within these frames. However, since the optical camera does not provide Cartesian positions as the Time of Flight camera does, it will only be used to determine a cross-range unit vector to the CubeSat centroid (see Fig. 10) for performing a correction in the $(\hat{v}_1, \hat{v}_2)$ frame, without performing a correction on the $\hat{v}_3$ value. These algorithms are broken into three main components. Each of these components will be discussed here.

For the image to be processed, the pixels of the image containing CubeSats can be logically defined. That is done by binarizing the image, and assigning values of logical true or false to each pixel, based on some threshold of visual magnitude. To perform this binarization, an adaptive thresholding method is used in order to keep the binarization threshold

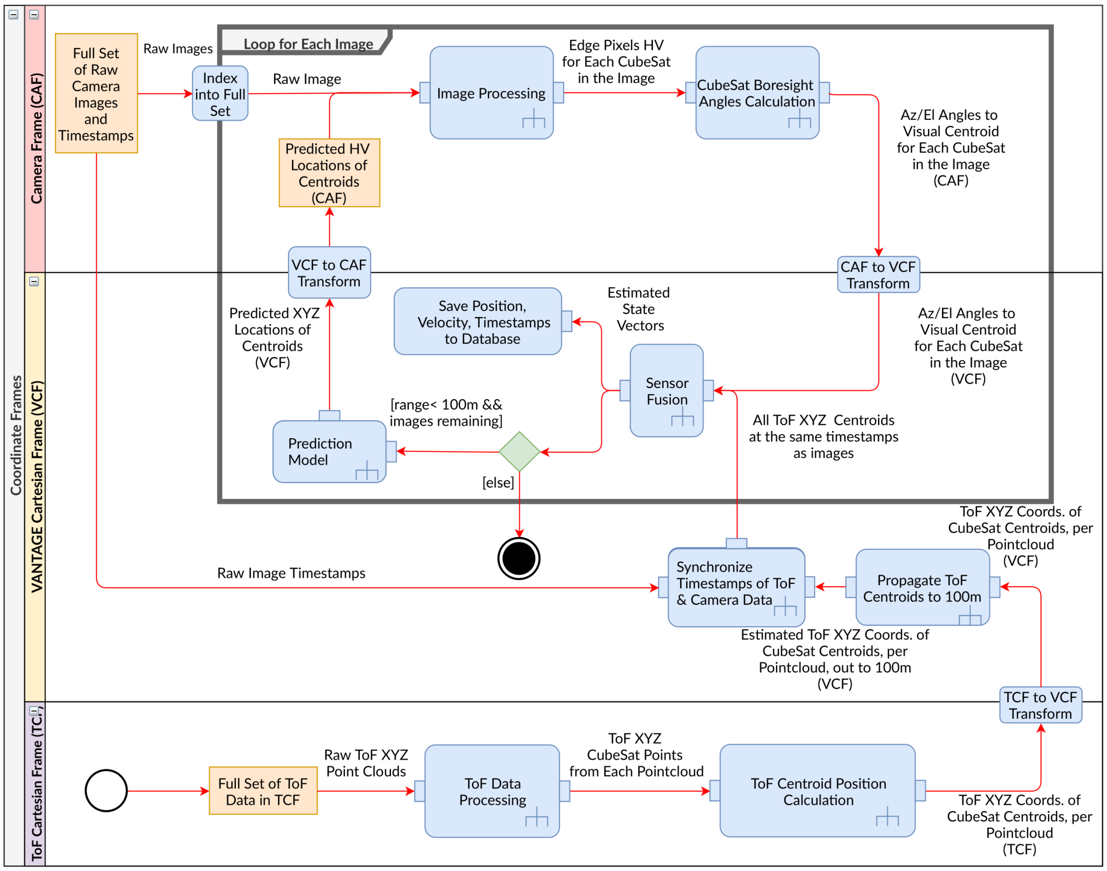

Figure 7: A UML Activity Diagram for the Post-processing Module of the VANTAGE Software

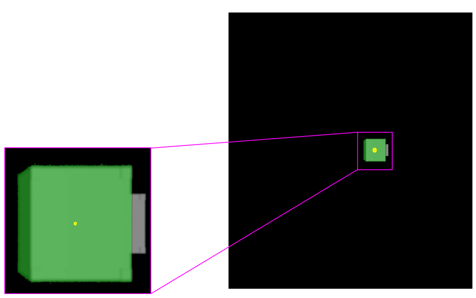

Figure 8: Example of the Results of the Algorithm to Determine and Segment Occluded CubeSats into Multiple Areas for Centroiding.

value consistent with the brightness of the CubeSats as they move down the range. First, a base threshold is assigned to the image which is known to be darker than any CubeSat pixels. Next, each frame is analyzed individually for the distribution of visual magnitude values. A new adaptive threshold for that image is then applied based on the distribution, eliminating the background noise below the frame's individual adaptive threshold. Upon completion, the vast majority of pixels remaining correspond to CubeSats.

The image processing software is designed specifically to handle multiple CubeSat occlusions that occur during deployment. Therefore, the software solution must be able to detect this, and extrapolate centroid values from previously known positions and velocities to continue obtaining results. Occlusion is detected based on the known geometry of how the CubeSats are deployed. In any case of occlusion, convex angles are formed at the outer edges of the CubeSats. The boundary pixels remaining after binarization are analyzed, and convex angles are marked that are detected around the boundaries. These detected angles allow for the shapes of the occluded CubeSats to be inferred by the software, as shown in Fig. 8, and these inferred boundaries are then used to compute centroid locations.

It is crucial that between images, centroids can be as-

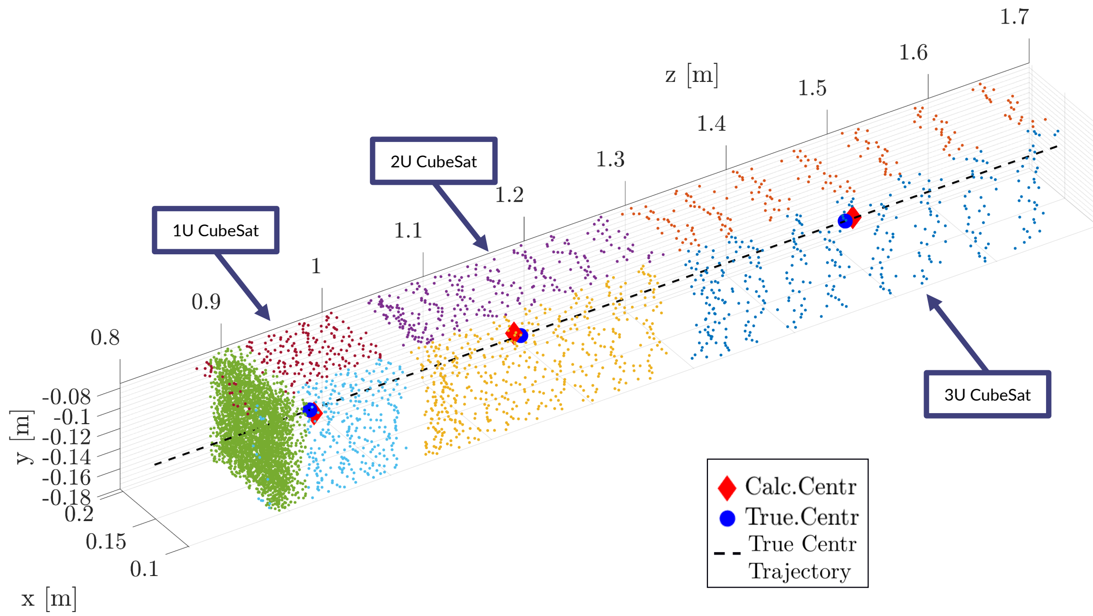

Figure 9: CubeSat Identification, Plane Fitting, and Centroid Calculation for a Simulated Deployment of Three CubeSats: (1U, 2U, 3U)

sociated to the same CubeSats as in previous frames. Therefore, the relative motion of the centroids is analyzed to ensure the object association remains consistent. Based on the placement of the optical camera relative to each deployment tube, it is known that during a deployment, centroids will all shift towards the centerline of the camera as deployment occurs. Therefore, object association can be performed by observing the distance to the centerline of each centroid relative to one another. This allows for consistent association of centroids to their respective CubeSats, as seen in Fig. 10.

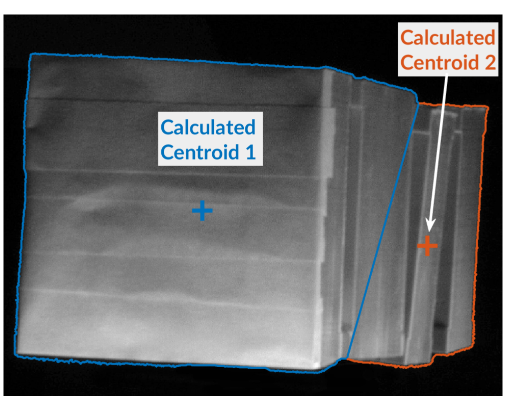

Figure 10: CubeSat Identification, Multi-object Tracking, and Centroid Calculation for a Simulated Deployment of Three CubeSats.

Reported State Vector Synthesis

Figure 11 details the method that is used for state vector synthesis (see Fig. 7 for the state vector synthesis's location in overall processing). Each sensor has its own native frame (TCF for ToF camera and CCF for monochrome camera). Reported state vector synthesis is run after the results from both sensors are converted to a common cartesian frame centered on the monochrome camera. As a simple explanation of the method, we find the line that is orthogonal to the camera vector and passes through the ToF estimated centroid. The synthesis point is on this line at a distance between the camera vector and ToF point that is determined by a weighting of the uncertainties for both centroid solutions. These uncertainties are initially calculated using the error tolerances reported in the sensor datasheets and then once implemented calibration can be done to get a more accurate approximation of the sensor result uncertainty as a function of range. This synthesis method returns positions in the VANTAGE Coordinate Frame (VCF) that are then used to calculate the velocity of the CubeSats.

Algorithm 1 An algorithm for determining 3D centroids of all CubeSats identified in a ToF point cloud file in the ToF sensor's coordinate frame.

1. Obtain point cloud from point cloud file
2. Convert the range measurement of each point in the point cloud to a smoothed probability density by computing the smoothed Kernel probability distribution function (pdf) of the range measurements.
3. Split point cloud into distinct CubeSats by identifying locations of local minimum point density in the pdf along the downrange axis
4. For each identified CubeSat:
   (a) Fit 1 to 3 planes to the faces of the CubeSat point cloud using a singular value decomposition (SVD) of the points
      i. Obtain face normal vectors
      ii. Ensure that face normals point into CubeSat by multiplying the normal vector by the sign of the dot product of the normal vector with a point in the plane
   (b) For each identified face:
      i. Smooth the boundary of the face using a Savitsky-Golay Filter
      ii. Find the minimum bounding box of the face and consider this to be the best estimate of the face
      iii. Find the centroid of the face by taking the mean of the corners of the minimum bounding box
      iv. Compare the face dimensions to the known CubeSat dimensions to determine the orthogonal distance from the face to the centroid
   (c) Find the centroid by one of the following:
      i. If one plane is identified, project from the face centroid along the normal vector by the orthogonal distance from the face to the centroid
      ii. If two planes are identified, project from the midpoint of the intersection of the faces along the normal vector of both faces respectively multiplied by the distance from the faces to the centroid
      iii. If three planes are identified, project from the intersection point of the three planes along the normal vector of all three faces respectively multiplied by the distance from the faces to the centroid

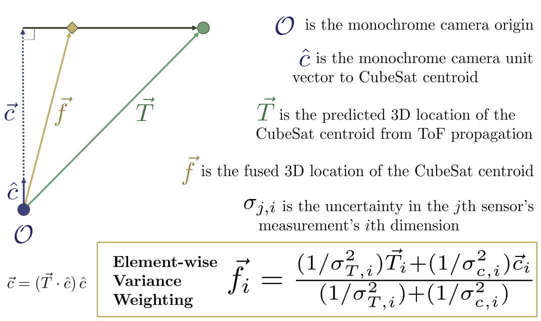

Figure 11: An Overview of the Mathematics of the Variance Weighting State Vector Synthesis Method Used.

TEST DESIGN

Validation of the VANTAGE design on the ground is important for determining the potential of this application in its final mission context. Three testing schemes are designed and implemented to verify the performance of the VANTAGE sensor package and software solution.

Overview of Testing Objectives

The testing objectives are centered around two fundamental questions.

Can a space-like deployment of the CubeSats be simulated on the ground for the 100m sensing range of the sensor package? To validate sensor package performance on the ground, 100m of simulated CubeSat trajectory is needed in order to examine the sensing range. The full 100m range can be split up into two different sections. The first 10m section is relevant to the ToF camera and the remaining 90m are more relevant to the Monochrome Camera. This range distinction is important in order to be able to verify the performance of each sensor as well as the full sensor package operating together.

Is the true position of the simulated CubeSats known well enough to be able to verify the performance of the sensor package? The truth position of the simulated CubeSats is used as the success criteria against

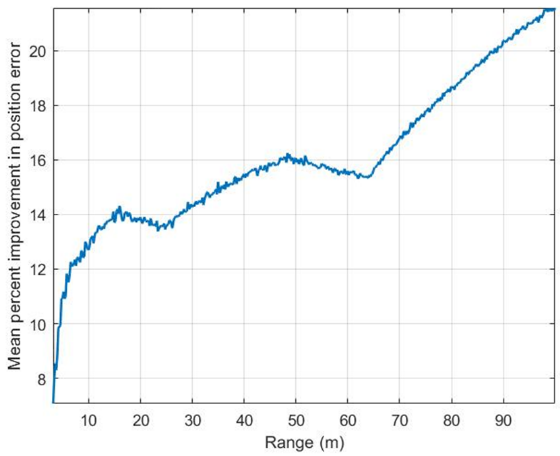

Figure 12: The Average Improvement on ToF State Measurements When Using the State Improvement Measurements from the Monochrome Camera

which the sensor and software solution's reported measurements of relative CubeSat position and velocity are verified. The measurement of these truth positions must be known to a least a factor of 10 better than the design requirements in order to verify the performance of the sensor package and software solution.

Based on these questions three independent testing schemes are implemented. The *Simulation Test* provides data sets over the full 100m range by creating simulated data outputs of both the ToF Camera and the Monochrome Camera in addition to perfect truth data measurements of the simulated CubeSat objects. The *Modular Test* is a physical test which focuses on the first 10m of the sensing range and ToF Camera verification. The third is the *Full System Test* which simulates the deployment of CubeSats from the ISS for the full 100m sensing range for beginning to end verification of the system.

Software Validation Through the Simulation Test

The Simulation Test is a method to do software validation of the VANTAGE software solution by running a predetermined deployment case through a simulation which generates both truth positions of the simulated CubeSat objects as well as simulated data for the ToF Camera and the Monochrome Camera. The objective of the Simulation Test is to provide initial verification of the CubeSat state calculation algorithms, accurately simulate the space environment, and mimic the sensor performance of the ToF and Monochrome Cameras.

The Simulation Test is also configured to allow for parametric testing of off-nominal deployment cases and other deployment cases difficult to simulate with the physical Modular and Full System Tests. The Simulation Test is capable of generating realistic data for any conceivable deployment configuration or deployment anomaly so long as it can be programmed into the powerful deployment framework.

The foundation of the Simulation Test is a combination of scripted Blensor and Cinema 4D simulations which are interconnected using a common file format. This framework relies on input configuration files which detail specific deployment configurations to be tested. Fig. 13 shows a frame of the results of one such simulation.

The programmable deployment framework of the Simulation Test makes it possible to easily vary a wide variety of parameters to build different deployment scenarios, control sensor specific settings and physical characteristics, as well as environmental conditions. This leads to robustness in the software solution because it can be tested with almost any conceivable deployment scenario. The Simulation Test parametric architecture is written using Python and YAML configuration files to allow for easy user configuration of different deployment cases to produce data sets for.

Cinema4D (C4D), which is the industry standard professional software for animation and effects rendering is used to simulate image capture of the Monochrome Camera and to animate and CubeSat objects in different deployment scenarios. C4D accurately simulates sensor properties (resolution and pixel size), lens properties (like focal length and aperture and the resulting Depth of Field (DoF)), environmental lighting, and material properties. These features produce images that accurately simulate the properties of the chosen Monochrome Camera, including many of its non-idealities.

The Blensor version of Blender™ is the only widely available ToF camera simulation software in existence. Blensor was created by two postdocs at the University of Salzburg to model basic properties of the chosen ToF Camera and more complex environmental effects like backscattering of IR rays from the ToF IR flash and sensor IR noise [6]. Blensor can import the C4D animation file of the deployment test case from Cinema 4D and then generate realistic

(a) Blensor Render of a Simulated ToF Point Cloud. The Point Cloud is Shown in Orange, Overlaid on Physical Models of the CubeSats.

(b) C4D Render of a Simulated Monochrome Camera Image. Taken from the same Simulation Timestep as the ToF Camera Point Cloud.

(c) Actual Monochrome Image Taken with our Sensor Suite During a 10m Modular Test.

Figure 13: Comparison of Simulated and Real Sensor Data

IR point clouds from the simulated CubeSat objects during the animated deployment at a desired sample rate.

The simulation framework also allows for significant expansion to refine the accuracy of hypothetical deployment cases. In real, physical deployment scenarios there are additional considerations such as non-linear dynamical motion (Clohesy-Wiltshire effects), irregular CubeSat geometries, and CubeSat angular velocities. The ability to expand the simulation beyond simple linear projections of the simulated payload means more life-like deployment scenarios could be generated and tested with the VANTAGE software solution.

Modular Testing

The Modular Test, shown in Fig. 14 is a physical test which simulates the motion of deployed CubeSats for the first 10m of the sensing range. This test is primarily used to characterize the performance of the ToF Camera in the sensor package. The ToF Camera is a critical part of the state estimation algorithm, thus it is important to have repeatable, iterable testing with an ultra-low error to ensure that the camera and corresponding software solution operate as expected. To achieve repeatability over a 10m range the Modular Test employs a 12m track on which a configurable cart with CubeSat shaped objects is pulled by a speed variable motor. The cart moves down the length of the track while the sensor package collects ToF and Monochrome image data. The configurable CubeSat objects are then positioned relative to the VANTAGE sensor package as they would be in their chosen deployment silo in the NanoRacks NRSCD assembly in the ISS. The true positions of the simulated CubeSats on the cart are tracked using a VICON system.

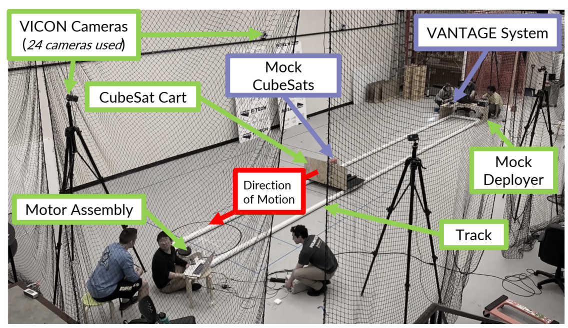

Figure 14: View of the Modular Test and VICON

The VICON system uses multiple infrared cameras to track IR markers placed on the cart. Their position is used to record the centroid of the cart as it travels down the track. The initial centroid location of each CubeSat object is known in relation to the cart centroid by measurement of a fixed offset vector. This offset position vector is used to relate the VICON measured cart centroid position to the CubeSat object centroid positions at all data collects during the test. This correlates the truth positions of the simulated CubeSat centroids to the positions measured by VANTAGE and allows for verification.

Full System Test

The Full System Test is designed to provide performance verification for the entire 100m sensing range and a platform to test the complete cycle of the VANTAGE system from initiation to data closeout as shown in the mission context diagram in Fig. 4. This test simulates the assumed linear motion of deployed CubeSats along the 100m sensing rang using a specially designed test rig (boom) which mounts to a motor vehicle. The vehicle provides steady, stable motion of simulated CubeSat objects for 100m of distance from the stationary sensor package which images the simulated CubeSat objects as the vehicle drives past and travels downrange. Fig. 15 shows the test rig extension boom mounted on the test vehicle. The test configuration of simulated CubeSats is mounted at the end of the extension boom. Just past the simulated CubeSats is a tripod holding the sensor package and the ground station which provides power and data connections simulating the ISS interface for the VANTAGE design. The test takes place along a long, straight path of flat ground located on at the Boulder Airport. A GPS unit is used to gather truth position data of the simulated CubeSats. A rigidly mounted Trimble NetR9 GNSS receiver with RTX option provides position truth data accurate to approximately 5cm for the simulated CubeSats. The offset vector from the Trimble GPS antenna to the simulated CubeSat objects is measured with a ruler in each test case in order to be able to compare the GPS truth data to the position estimates from the VANTAGE software solution.

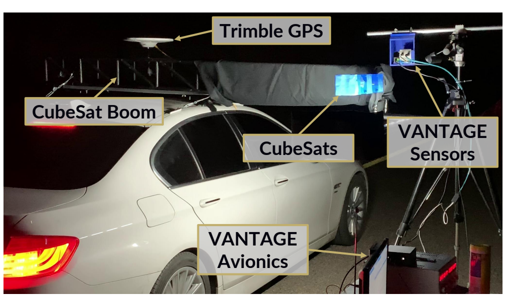

Figure 15: View of the Full System Test

RESULTS

The physical tests and simulation described above were successfully used to test and characterize the sensor package and software solution. The findings support software solution verification on a functional level with data generated from the Simulation Test framework as well as data obtained from the Modular and Full System Test. Fig. 16 shows the resulting absolute position error of the CubeSat positions returned from the software solution for simulation data, Modular Test data, and Full System Test data.

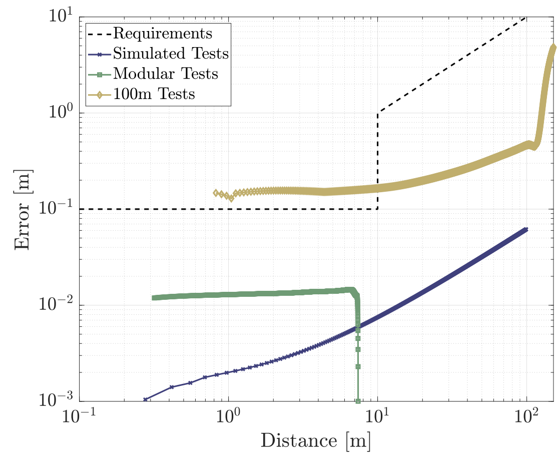

Figure 16: Average 3D Absolute Position Error for All Three Test Systems

Simulation Results

The absolute position errors of the positions reported by the software solution when fed simulation data are smaller than for the physical tests. The simulation environment removes uncertainties in angular offsets between coordinate frames and produces truth position data without uncertainty. These two characteristics contribute to the lower overall absolute position error associated with the reported measurements when run on data from simulated deployments.

Empirical Results

The physical tests resulted in 20 sets of Modular Test data for a single deployment case and 10 sets of Full System Test data for another single deployment case. Since the empirical tests introduce additional uncertainties resulting from noise, and additional systematic biases due to alignment, multiple runs of identical deployment scenarios were tested in order to characterize the performance of the sensor payload and software solution. In both cases the average systematic bias of the system is determined and corrected for resulting in the performance shown in Fig. 16.

The truth position data obtained in the Full System Test is correlated to each deployment case tested using a series of coordinate transforms derived from empirical measurements between the orientation of the sensor payload and the measurement frame of the truth position data. For the Modular Test the truth positions of the CubeSats and the position of the sensor payload are measured in the same coordinate frame by the VICON system and thus only the transform of the sensor payload orientation to the VICON system origin is measured and applied.

The Modular Test data runs are then averaged together to determine the performance of the sensor packages and software solution. The same is done for the Full System Test resulting in the curves shown in Fig. 16. As expected the absolute position errors reported for the physical tests are larger because of the added uncertainty in the truth position measurements and the systematic biases in sensor alignment at the test. Overall, performance of the sensor packager and software solution developed for this application yield results with absolute position errors below the targets set forth in the *Design Objectives* section.

Future Work

The findings presented above begin to validate the proposed sensor system design software solution using both high fidelity simulations and physical tests. Completion of this verification process for this ground based design is the first step towards implementing design considerations for a space-ready system. Once the verification of the initial design concept presented in this paper is complete, then work will likely take two different paths which may be worked in parallel. The first of these paths is further iterations of the software solution. The most readily accessible path towards improving the fidelity, accuracy, and applicability of this system is to continue to refine the sensor package capabilities and the software algorithms doing the calculations. The second of these paths is work towards implementation of the sensor and software solution on space-rated hardware. This implementation is not without challenges and will certainly be dependent on the complexity and sensitivity of later iterations of the system design.

ACKNOWLEDGEMENTS

The authors would like to acknowledge the contributions of all the additional VANTAGE team members: Marshall Herr, Dylan Bossie, Justin Fay, Sean Downs, Jiarui (Jerry) Wang, Zach Talpas, Richard Moon, and Josh Kirby. We appreciate support from the University of Colorado Boulder, the Ann & H.J. Smead Department of Aerospace Engineering Sciences and, Cu RECUV Lab as well as the members of the Senior Projects Advisory Board, project advisors Prof. Penny Axelrad, Prof. Marcus Holzinger, and John Gaebler. We acknowledge NanoRacks LLC for their technical contributions to the project and for supplying hardware for verification. The Full System Test portion of this project relies on equipment provided by the UNAVCO Facility with support from the National Science Foundation (NSF) and National Aeronautics and Space Administration (NASA) under NSF Cooperative Agreement No. EAR-0735156.

References

[1] Goparaju Sudhakar. Isro: 104 satellites in 1 go. In *Vidyaniketan Journal of Management Research*, pages 74–94, 03 2018.

[2] T.S. Kelso. Challenges identifying newly launched objects. In *68th International Astronautical Congress*, 09 2017. IAC-17.A6.7.9.

[3] J. A. Gaebler and P. Axelrad. Extracting cubesat relative motion using in-situ deployment imagery. In *42nd AAS Annual Guidance and Control Conference, Breckenridge, CO*, 02 2019. AAS 19-016.

[4] Paul W. Schumacher, John Gaebler, Christopher W. T. Roscoe, Matthew P. Wilkins, and Penina Axelrad. Parallel initial orbit determination using angles-only observation pairs. *Celestial Mechanics and Dynamical Astronomy*, 130, 09 2018.

[5] NanoRacks. Nanoracks cubesat deployer (nrcsd) interface definition document (idd). 2018. NR-NRCSD-S0003.

[6] Michael Gschwandtner. *Support Framework for Obstacle Detection on Autonomous Trains*. PhD thesis, Department of Computer Sciences, University of Salzburg, 2013.