REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*
04-19-2017

2. REPORT TYPE
Journal Paper

3. DATES COVERED *(From - To)*
1 Jan 2014 - 1 Mar 2016

4. TITLE AND SUBTITLE
Nanosatellite Maneuver
Planning for Point Cloud
Generation With a Rangefinder

5a. CONTRACT NUMBER
N/A

5b. GRANT NUMBER
N/A

5c. PROGRAM ELEMENT NUMBER
N/A

6. AUTHOR(S)
Michael Nayak, Bogdan Udrea, Jaclyn Beck

5d. PROJECT NUMBER
N/A

5e. TASK NUMBER
N/A

5f. WORK UNIT NUMBER
N/A

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Air Force Institute of Technology, 2950 Hobson Way, Wright Patterson AFB, OH 45433
Embry-Riddle Aeronautical University, 600 S Clyde Morris Blvd, Daytona Beach FL 32114
NASA Goddard Space Flight Center, 8800 Greenbelt Rd, Greenbelt MD 20770

8. PERFORMING ORGANIZATION REPORT NUMBER
N/A

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Air Force Institute of Technology, 2950 Hobson Way, Wright Patterson AFB, OH 45433

10. SPONSOR/MONITOR'S ACRONYM(S)
AFIT/CIP

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
N/A

12. DISTRIBUTION/AVAILABILITY STATEMENT
Distribution A. Approved for public release: distribution unlimited.

13. SUPPLEMENTARY NOTES
IEEE Transactions on Aerospace and Electronic Systems 51 (4), 3085-3098; DOI. No. 10.1109/TAES.2015.140075.

14. ABSTRACT

This paper discusses the application of a single beam laser rangefinder (LRF) to point cloud generation, shape detection, and shape reconstruction for a space-based space situational awareness (SSA) mission. The LRF is part of the payload of a chaser satellite tasked to image a resident space object (RSO). The one-dimensional (1D) nature of LRF returns significantly increases the complexity of the imaging task. To maximize coverage, a method to autonomously detect and fill gaps in sparse point cloud coverage using a narrow field of view (NFOV) camera in conjunction with the LRF is presented. Maneuvers can now be designed on-line in real time; results presented validate the utility of a single-beam LRF as a tool for 3D imaging of RSO shapes.

15. SUBJECT TERMS
Nanosatellite, laser rangefinder, formation flying, USAF, object imaging, shape reconstruction, space-based, space situational awareness, active debris removal, small satellite

16. SECURITY CLASSIFICATION OF: a. REPORT	b. ABSTRACT	c. THIS PAGE	17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
UU	UU	UU	NONE	14	MICHAEL NAYAK

19b. TELEPHONE NUMBER *(Include area code)*
808-891-7727

Reset

Nanosatellite Maneuver Planning for Point Cloud Generation With a Rangefinder

MICHAEL NAYAK
University of California
Santa Cruz, CA, USA

JACLYN BECK
NASA Goddard Space Flight Center
Greenbelt, MD, USA

BOGDAN UDREA
Embry-Riddle Aeronautical University
Daytona Beach, FL, USA

This paper discusses the application of a single beam laser rangefinder (LRF) to point cloud generation, shape detection, and shape reconstruction for a space-based space situational awareness (SSA) mission. The LRF is part of the payload of a chaser satellite tasked to image a resident space object (RSO). The one-dimensional (1D) nature of LRF returns significantly increases the complexity of the imaging task. To maximize coverage, a method to autonomously detect and fill gaps in sparse point cloud coverage using a narrow field of view (NFOV) camera in conjunction with the LRF is presented. First, relative orbital motion and scanning attitude motion are combined to generate a baseline 3D point cloud of the RSO. The effectiveness of pregenerated command profiles is analyzed by using a weighted edge reconstruction metric that scores how well a point cloud characterizes RSO shape. The design and characterization of multiple relative motion and attitude maneuver profiles, as well as the time and propellant cost of each profile, are presented with the assumption that the entire metrology chain is error free. Next, a three-part algorithm is used that 1) creates a 3D panoramic map from stitched NFOV camera images, 2) correlates the areas of sparse LRF coverage to the map, and 3) generates attitude commands to close the coverage. This provides a consistent and reliable method to register positions of strike points relative to each other and to the NFOV image of the RSO with a priori knowledge of the RSO attitude. Gaps and sparse areas in LRF coverage are covered with strike points; the result is a point cloud of significantly higher resolution than that obtained with preprogrammed attitude profiles. The analysis bears particular relevance to power-constrained nanosatellite missions for space-based SSA for whom a multibeam LRF payload is not feasible. Maneuvers can now be designed on-line in real time; results presented validate the utility of a single-beam LRF as a tool for 3D imaging of RSO shapes.

I. INTRODUCTION

Rendezvous and proximity operation [1–4] missions as well as future small body exploration missions such as NASA's OSIRIS-REx [5–7], rely on three-dimensional (3D) imaging of man-made or natural resident space objects (RSOs). The Application for RSO Automated Proximity Analysis and IMAging (ARAPAIMA) mission is a proximity operations mission sponsored by the Air Force Research Laboratory to perform the in-orbit demonstration of visible, infrared (IR), and 3D imaging of RSOs from a nanosatellite platform [8–10]. To accomplish the objectives from a nanosatellite platform, the ARAPAIMA mission hosts a simple payload consisting of a narrow field of view (NFOV) monochrome camera and a laser rangefinder (LRF).

Previous work in the literature discusses obtaining real-time relative pose estimates in a proximity operations scenario using laser-aided active vision systems [11], dense stereo [12], and TriDAR [13]. However, these systems are unsuitable for a nanosatellite system from power, size, and on-board computation points of view. Approximately the size of a shoebox, nanosatellites are intrinsically power constrained [14]. This is especially true for a proximity operations mission such as ARAPAIMA, which requires precision propulsion control and an IR imager for navigation during the eclipse side of the orbit [15]. Power requirements do not allow for a multiple-beam LRF; the need for all payload components to fit within a 6U CubeSat form factor [16] and the desire to maximize use of commercial off-the-shelf hardware and minimize mechanical complexity do not allow for a gimbaled LRF. The resulting 1D nature of LRF measurements increases the complexity of the imaging task significantly [17]. As a consequence, the application of an LRF to generate a point cloud of an RSO with sufficient resolution to provide mission-capable registration and shape reconstruction is highly challenging.

This article presents a method that uses the NFOV camera payload to supplement LRF data and autonomously detect and fill gaps in sparse point cloud coverage. Positions of strike points are registered relative to each other and to the NFOV image of the RSO, using a priori knowledge of the RSO attitude. This method builds upon previous work by the authors to introduce various preplanned translation and attitude command profiles as well as automated coverage gap detection and gap filling

Manuscript received January 29, 2014; revised November 2, 2014, May 15, 2015, June 4, 2015; released for publication June 5, 2015.

DOI. No. 10.1109/TAES.2015.140075.

Refereeing of this contribution was handled by M. R. Akella.

M. Nayak would like to acknowledge funding from Red Sky Research, LLC. This research was also partially conducted with government support to M. Nayak under and awarded by DoD, Air Force Office of Scientific Research, National Defense Science and Engineering Graduate (NDSEG) Fellowship, 32 CFR 168a.

Authors' addresses: M. Nayak, Department of Earth and Planetary Sciences, University of California Santa Cruz, CA 95060, E-mail: (mnayak@ucsc.edu); J. Beck, NASA Goddard Space Flight Center, Greenbelt, MD 20771; B. Udrea, Department of Aerospace Engineering, Embry-Riddle Aeronautical University, Daytona Beach, FL 32114.

---

algorithms that can produce point clouds of sufficient fidelity for shape reconstruction [18–21]. The improvement over preprogrammed attitude profiles is quantified. Ultimately, with the demonstrated ability to generate attitude command profiles on-line and potentially in real-time to fill in coverage gaps, simple mission planning and relatively high-resolution 3D point cloud generation can be implemented in onboard flight software.

Section II briefly reviews an edge metric developed to evaluate the quality of LRF point clouds generated by various motion command profiles. Sections III–V describe the algorithmic generation of the profiles and which types of profiles are surveillance time and propellant efficient. Sections VI–VIII describe the combination of camera images with LRF measurements to perform autonomous gap detection using panoramic mapping and Voronoi diagrams. Results and conclusions are drawn in Sections IX and X.

II. EDGE METRIC

Previous work by the authors [18] details an edge metric independent of the size of the point cloud, the size of the target RSO size, and number of edges of the RSO. The metric quantifies the ability of the 3D shape reconstruction algorithm to process the LRF measurements into edges of the original shape and assigns a score based on the relative magnitude of the errors, by comparison against the known-truth model. Each point in a point cloud is a measurement taken by the LRF, and it represents a position vector in a Cartesian reference frame attached to the chaser. Using the surface variance, the raw point cloud is deconstructed into clusters that are likely to be along an edge of a shape [22]. Once a collection of candidate edge points is found, points are separated into groups belonging to the closest known-truth edge in the model and a line is fit to each group of edge points. The estimated line for each group is scored on three baselines.

1) Angle score is calculated from the dot product between the fitted line's direction and the true edge. The result is then subtracted from 1, i.e., larger angles yield larger error.
2) Offset score is the distance between the midpoint of the fitted line and the midpoint of the true edge. A smaller offset yields lower scores and measures how much the line is offset from the true edge.
3) Cover length score measures how much of the true edge is covered by the fitted line. The absolute difference between the length of the true edge and length of the fitted line is calculated. Larger discrepancies in length give higher scores.

The combined edge metric score is the simple average of the three scores. The closer the score is to zero, the better the fidelity of the point cloud in recovering the shape of the RSO and its edges.

To obtain a baseline for the metric, multiple point cloud density simulations were run with an RSO (Fig. 1) in a circular low Earth orbit (LEO) at an altitude of 400 km, an initial nadir pointing attitude, and constant slewing of 1° in either direction along its z-axis. The RSO body is a 5 m cube, measuring 20 m between solar panel ends. Two additional RSO appendages have been modeled: a circular disk of finite thickness serves as a model for an open cover for imaging payloads and a sphere serves as a model for a sensor pod. It is expected that capturing the shape of the flat cover with a 1D LRF will be challenging; reconstruction of this appendage is the chosen marker between an acceptable and unacceptable point cloud. The LRF pulse repetition frequency (PRF) was varied between every 0.1 and 2 Hz in steps of 0.1 Hz to create point clouds of increasing size. The maximum simulation time was fixed at 36 h (~28 orbits) regardless of point cloud size. Fig. 2 shows edge points (left) and reconstructed RSO shapes (right) for three sample simulations.

In Fig. 2, case (a) is an example of a very good reconstruction; case (b) is a limiting case between good and poor reconstruction, with definition of the flatter protrusion beginning to fall off; and case (c) is an example

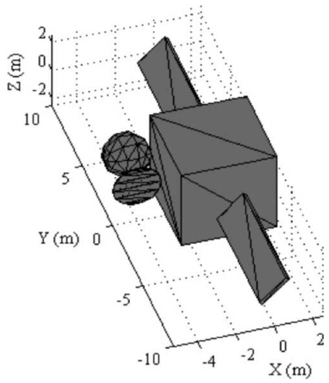

Fig. 1. RSO with protrusions used in simulations.

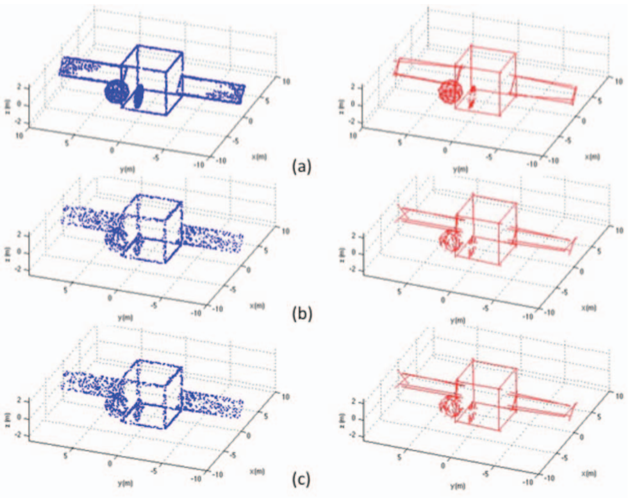

Fig. 2. Comparison of edge reconstructions of RSO from point clouds of various sizes. (a) Point cloud size: 89 511; score: 0.1236; (b) Point cloud size: 33 306; score: 0.3028; (c) Point cloud size: 11 241; score: 0.4453.

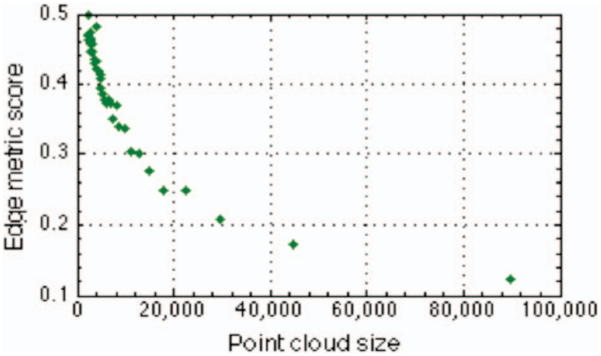

Fig. 3. Logarithmic relationship between edge metric score and point cloud size.

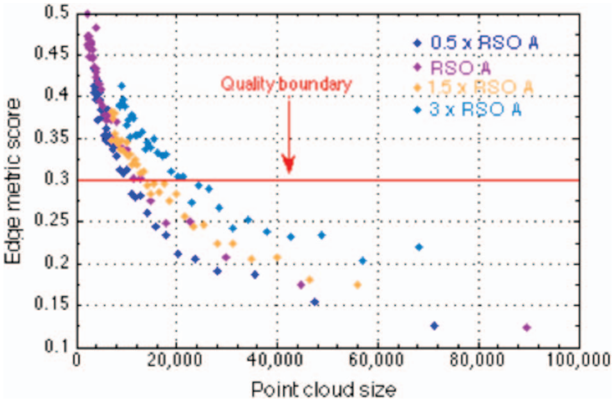

Fig. 4. Edge metric score versus point cloud size for four different RSO sizes.

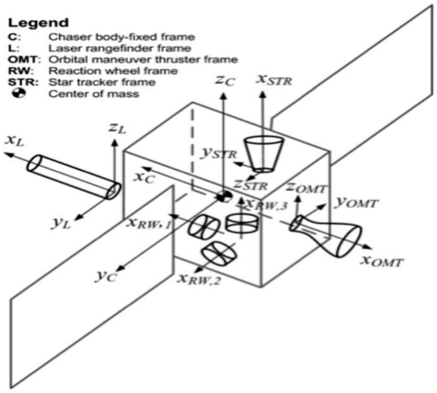

Fig. 5. Illustration of chaser satellite-based reference frames.

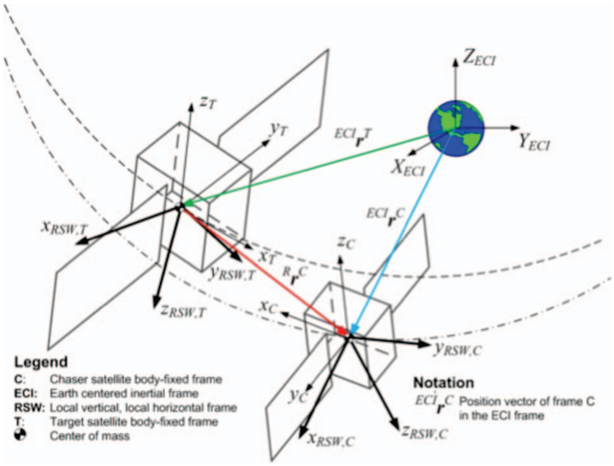

Fig. 6. Satellite reference frames with respect to Earth-centered inertial frame.

of a cloud that is too sparse, with neither protrusion successfully resolved by the LRF. Fig. 3 shows the logarithmic relationship between the metric score and the point cloud density. The strong correlation between the size of the point cloud and the metric score suggests the definition of a threshold for reasonable shape resolutions. For the size of the RSO analyzed here, the threshold is a score of 0.3, which corresponds to a point cloud size of 10 000 points. It is still possible to recognize basic shapes and outlines with the naked eye on sparse clouds such as case (c); however, for close range autonomous proximity operations, the shape shown in case (b) would be a minimum requirement.

The next question is the dependence between metric score and the size of the RSO. The size was varied by scaling the dimensions of the model in Fig. 1 by 0.5, 1.5, and 3; the results are shown in Fig. 4. The minimum point cloud density required to obtain acceptable shape reconstruction increases as size increases. However, because the edge metric score is invariant of size, 0.3 is a good indicator of the minimum point cloud density required. To obtain reliable edge recognition for the RSO shown in Fig. 1, with scale factors of 0.5, 1, 1.5, and 3, approximately 9000, 10 000, 13 000, and 20 000 points are needed, respectively. The results can be extrapolated to other RSO sizes as needed.

Based on this evidence, an empirical rule is employed; any point cloud that scores below 0.3 is sufficiently dense for accurate shape recognition, regardless of RSO shape or size. It is likely that the score threshold is slightly different for various shapes of RSO; however, it is expected that the logarithmic trend does not change.

III. DESIGN OF RELATIVE ORBITS

A. The Three Point Boundary Problem

For an RSO in a circular orbit and a chaser satellite separated from it by a distance significantly less than the orbit radius, the equations of motion are linearized to allow for simple maneuver design and analysis. They are called the Hill-Clohessy-Wiltshire equations (HCW) and are shown in (1). The reference frames used for relative orbit and maneuver planning are illustrated in Figs. 5 and 6. The fundamental plane of the RSO is defined by the S and W axes of the RSW reference frame. The RSW frame has the origin at the center of mass of the RSO, *R* is the unit vector along the RSO position vector, *W* is the unit vector along the orbit normal, and *S* completes the right handed system. The RSO fundamental plane is shown as a gray surface in Fig. 9 and subsequent figures.

$$x\,(t) = (4 - 3\cos nt)\,x_0 + \frac{\sin nt}{n}x_0 + \frac{2}{n}(1 - \cos nt)\dot{y}_0$$

$$
\begin{aligned}
y(t) &= 6(\sin nt - nt)x_0 + y_0 - \frac{2}{n}(1-\cos nt)x_0 \\
&\quad + \frac{4\sin \dot{n}t - 3nt}{n}\dot{y}_o \\
z(t) &= z_0 \cos nt + \frac{\dot{z}_0}{n}\sin nt \\
\dot{x}\,(t) &= \dot{x}_0 \cos nt + (2\dot{y}_0 + 3n\dot{x}_0)\sin nt \\
\dot{y}\,(t) &= (6nx_0 + 4\dot{y}_0)\cos \dot{n}t - 2\dot{x}_0 \sin nt - (6nx_0 + 3\dot{y}_0) \\
\dot{z}\,(t) &= \dot{z}_0 \cos nt - z_0 n \sin nt
\end{aligned} \tag{1}
$$

where $n = \sqrt{\mu/a^3}$ is the mean motion, i.e., angular rate of the orbit. The secular term $(6nx_0 + 3\dot{y}_0)$ of the $y(t)$ equation causes the y coordinate to drift over time. To maintain a stable relative orbit, initial conditions must be chosen so that the effect of the secular term is canceled; otherwise, periodic relative orbit adjustments must be performed. The literature describes manipulating and bounding the HCW equations [23, 24]; using this, a closed, nondrifting orbital path can be found that includes any two points in space, called waypoints [25]. Employing the waypoint method allows flexibility in path selection; relative orbital paths can be propagated between arbitrary numbers of waypoints. An added benefit is the applicability of waypoints to imaging appendages such as an antenna or telescoping boom that have primary dimensions either along a plane or one primary axis. Constraining the chaser to a single orbital path yields extremely poor coverage due to the 1D nature of the LRF measurements; this flexibility is crucial to the generation of 3D point clouds. However even if connectable near the local horizontal, relative orbits connecting waypoints may drive large eccentricity changes or pull the chaser too far away from its target. Limitations on waypoint selection are considered below.

B. General Limitations on Waypoint Selection

While there are many ways to choose waypoints, a relative orbit with some desired orbital elements, such as RAAN (right ascension of the ascending node), argument of periapsis, and inclination, is generated first. A certain set of waypoints on the relative orbit is then selected for propagation. For example, if two waypoints are selected on a relative orbit with an RAAN of 90° and either a 60° or 120° inclination, the orbit is circular. When transitioning from a given set of waypoints $\{w_1, w_2, \ldots, w_k\}$ to a subsequent set $\{w_{k+1}, w_{k+2}, \ldots, w_n\}$, it is desirable for the transition to occur at such a waypoint that minimizes the change in velocity and hence the propellant required to transfer to the new trajectory. However, depending on the location of the transfer waypoint, the path with the lowest transfer ΔV cost may be highly convenient (Fig. 7a) or impractical (Fig. 7b).

Relative orbits may bring the chaser dangerously close to the RSO or too far away for effective surveillance. Fig. 8 shows an example of poorly chosen waypoints that result in an orbital path that takes the chaser several tens of kilometers away from the RSO, diminishing the LRF's ability to collect a wide swath of points. Examples of poorly chosen waypoints are in the cross-track direction from the other waypoints, or ≤20° separation from a previous waypoint, as measured from the RSO fundamental plane.

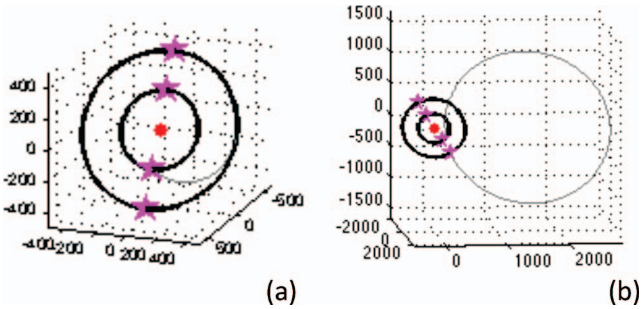

Fig. 7. Two waypoint cycles (black) connected by lowest ΔV cost path (grey). (a) is short time reconnection path, while (b) is long reconnection path.

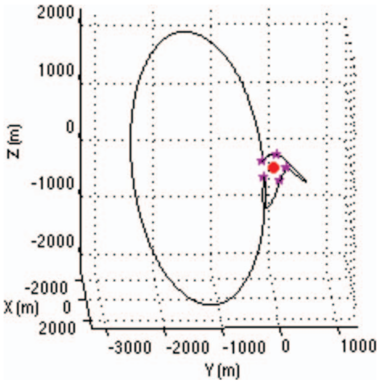

Fig. 8. Poorly chosen waypoints result in high ΔV costs and increasing distance from RSO (red dot) due to natural dynamics.

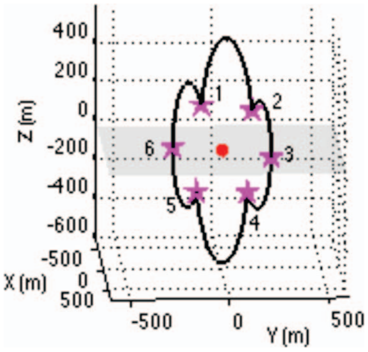

Fig. 9. Equidistant waypoints on polar circular surveillance orbit. Gray planar surface is RSO fundamental plane.

Fig. 9 shows six waypoints chosen on a circular orbit at an 83° inclination to the RSO fundamental plane, but the HCW-propagated paths the spacecraft must take to reach them cause sharp changes in direction. The ΔV of one orbit in this waypoint cycle is approximately 224 cm/s. However, if the four points at the sharp turns are moved just slightly out of plane, the ΔV cost is reduced by 85% to 33cm/s. In general, nearly polar relative orbits have much higher ΔV costs than orbits at lower

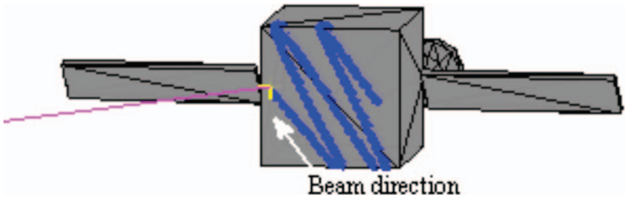

Fig. 10. Striping attitude profile. White arrow denotes beam direction.

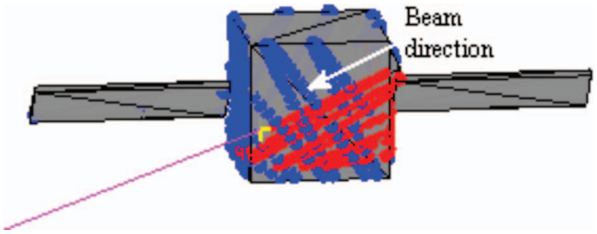

Fig. 11. Crosshatching attitude profile. White arrow denotes beam direction.

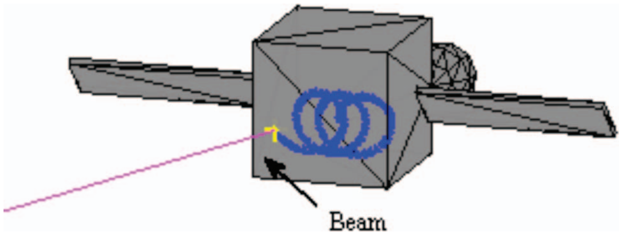

Fig. 12. Spiraling attitude profile. Black arrow denotes beam direction.

inclinations. It is more propellant-efficient to choose waypoints on paths that travel over the poles (+Z and –Z axes) but do not belong to a consistent, coplanar orbit waypoint path. Conversely, relative orbits chosen near the local horizontal (RAAN ~0°) will have high relative orbit eccentricities and consequently result in increased propellant consumption. The following section illustrates the points stated above with various examples of relative motion profiles.

IV. MANEUVER COMMAND PROFILES

A maneuver command profile is defined as any combination of waypoints, attitude maneuvers, and propulsive maneuvers that will be executed by the chaser in a certain amount of time. A near-optimal command profile utilizes a minimum amount of propellant to achieve a desired score for RSO reconstruction. Three types of attitude maneuvers, two types of in-plane propulsive maneuver profiles, and three types of out-of-plane propulsive maneuver profiles have been implemented; they are outlined below and shown in Figs. 10–12.

A. Attitude-Only Maneuvers

1) *Striping*: The chaser slews back and forth about one axis only, which has the effect of imparting a sweeping up-and-down motion of the beam. This maneuver allows for more widespread coverage than an RSO center-pointing beam.

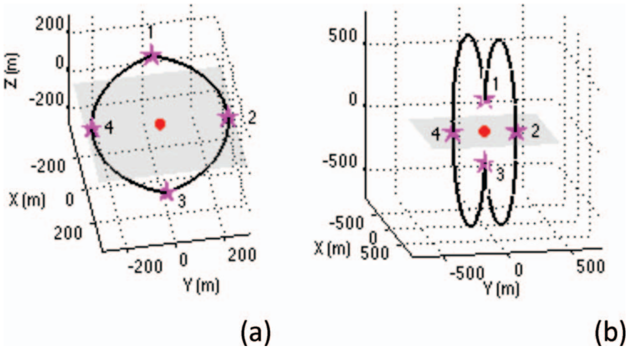

Fig. 13. ISOP command profile.

2) *Crosshatching*: The chaser alternates between striping about two perpendicular axes. This maneuver helps fill in gaps left by striping alone.

3) *Spiraling*: The chaser attitude is commanded such that the LRF beam describes a cone, creating spirals of strike points on the RSO.

For the RSO considered here, it is found that crosshatching with a slew angle of ±1° is most useful for obtaining a wide area of coverage. However, spiraling with a slew angle of ±0.5° covers areas of interest on the RSO body with relatively high fidelity. Spiraling with a larger slew angle, of ±1°, better covers relatively wide RSO appendages such as solar panels and protrusions. Therefore, all test cases implement crosshatching until a certain point cloud density is reached, followed by spiraling to gather better resolution.

B. Propulsive Maneuvers

1) *In-Plane Single Orbital Path (ISOP)*: Uses three or more waypoints to force a single orbital path with a shape, such as a straight line, that is not a conic section arc between waypoints. Fig. 13 shows four waypoints in an ISOP profile, chosen to force a somewhat circular orbit, at a relative inclination of 45° (Fig. 13a) and 85° (Fig. 13b). As the inclination of the plane with respect to the RSO reference plane increases, waypoints deviate further from the natural conic section arc, increasing the ISOP ΔV. In Fig. 13, the chaser must change direction sharply at waypoints 1 and 3, and thus these maneuvers have a ΔV cost of 41 cm/s and 320 cm/s, respectively, incurred every orbit.

2) *In-Plane Coorbital Transfer (ICOT)*: Uses two waypoints to create a natural ellipse arc and then makes a transfer to a different in-plane ellipse defined by two separate sets of waypoints. The four waypoints shown in Fig. 14 are the same with those shown in Fig. 13a, but the difference between ISOP and ICOT is evident. The black ellipse is defined by waypoints 1 and 4 from Fig. 13a, and the grey ellipse is defined by waypoints 2 and 3. The transfer path is created by using the chaser position on one path as a temporary waypoint and propagating to a waypoint on the second path. The ΔV cost is the cost of

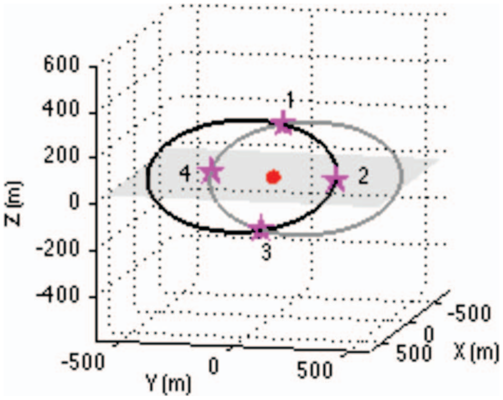

Fig. 14. ICOT command profile.

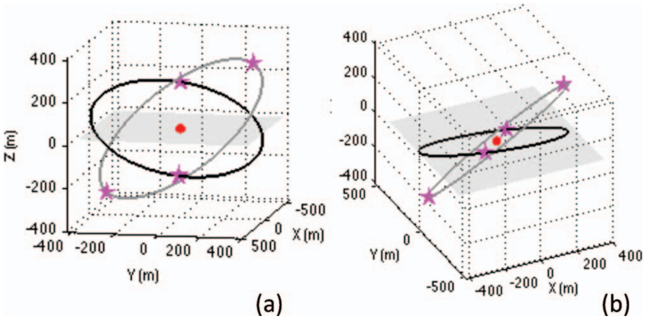

Fig. 15. OTAP command profile.

the transfer, ~23 cm/s for Fig. 14 and is incurred only once per orbit.

3) *Out-of-Plane Single Orbital Path (OSOP)*: All the waypoints used are part of a single path but with one or more out of plane. Fig. 9 shows how moving a carefully selected waypoint even slightly out of the desired orbital plane has the potential to reduce ΔV. The ΔV to maintain the OSOP path is incurred every orbit.

4) *Out-of-Plane Transfer of Argument of Periapsis (OTAP)*: Creates two ellipses with different arguments of periapsis and contains one transfer between the paths. To change the argument of periapsis of an HCW-based ellipse, the RAAN must change as well. This results in two ellipses in two separate planes, with the transfer in a third plane, as shown in Fig. 15a (front view) and Fig. 15b (bottom view). The ΔV cost to transfer between the ellipses shown is 33 cm/s and is incurred once per orbit.

5) *Out-of-Plane Inclination Transfer (OPIT)*: Creates two ellipses at different inclinations with respect to the target reference plane and one transfer between them. The ΔV cost of this profile is equal to the cost of the transfer and is incurred only once per orbit.

In conclusion, longer paths and smooth changes are preferred. Intermediate paths are better than direct transfers that perform sharp relative orbital parameter changes. Next, the combinations of these attitude and propulsive profiles are compared and ranked. The primary metric will be the ΔV consumed to reach a threshold point cloud density of 0.3.

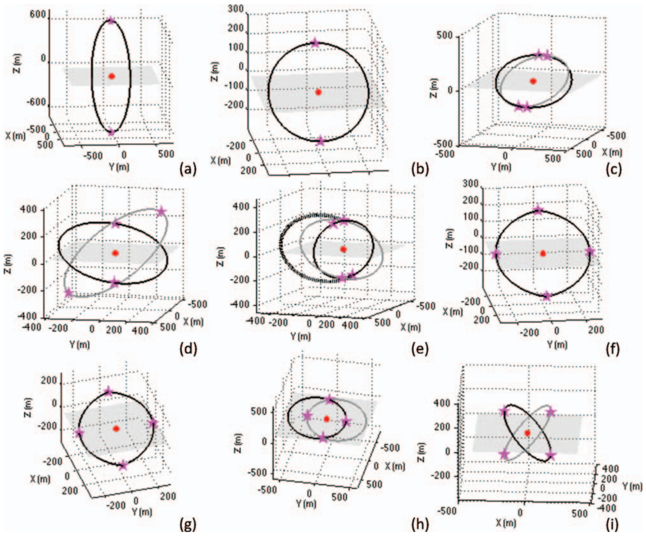

Fig. 16. Nine test command profiles. Fundamental plane is highlighted in grey. In (e), transfer orbit is dotted line.

V. TEST CASES

Nine test cases are presented in Fig. 16. They are designed to assess different relative orbital path and attitude maneuver combinations. The results of their effectiveness have been ranked by edge metric score of the point cloud and the ΔV cost. The evolution of the score over time for each test case is presented in Table I to judge how fast a given profile is able to obtain an edge metric score below 0.3.

Coverage time is capped at 36 h. It is assumed that the chaser employs reaction wheels as the actuators of the attitude control subsystem, and the attitude profiles presented here have been designed to avoid their saturation. A slew half period of 40 s is implemented for both maneuvers. The LRF is pulsed once every second (PRF 1Hz) and the chaser satellite follows each path for a specific amount of time. All times noted are in hours. In all cases, the chaser performed crosshatching with a slew angle of $\pm 1^\circ$ for exactly half the simulation time and spiraling with a slew angle of $\pm 0.5^\circ$ for the other half. Simulations that involve two paths follow this same convention for each individual path.

VI. SPATIALLY RELATING LRF STRIKES THROUGH PANORAMIC MAPPING

A. Camera-Assisted Point Cloud Generation

Preprogrammed attitude and maneuver profiles between arbitrary waypoints described in the previous section are stepping-stone for expansion into autonomous point cloud generation. The edge metric score tends to level off or stagnate if the chaser continues using the same preplanned command profile [18]. Therefore, there is a need for the chaser to be able to autonomously generate its own attitude and maneuver profiles to fill in sparse coverage areas of the point cloud it has collected thus far, increasing the detection score and collecting new

TABLE I
Score after 2 h, Time to 0.30 Score, and Details of Test Cases in Fig. 16

	Maneuver	Score After 2 hr	Delta-V Cost (cm/sec)	Rep	Time to 0.30 Score (hrs)	Notes
Case 1	No prop	0.3481	0	N/A	> 36	2 waypoints, RAAN = 90, Inc = 80
Case 2	No prop	0.3297	0	N/A	> 36	2 waypoints, RAAN = 90, Inc = 60
Case 3	OTAP	0.2928	17.6538	One-time	12	Path 1 (dark): 2 waypoints, RAAN = 90, Inc = 50 Path 2 (light): 2 waypoints, RAAN = 60, Inc = 50
Case 4	OTAP	0.2672	33.2368	One-time	12	Path 1 (dark): 2 waypoints, RAAN = 60, Inc = 50 Path 2 (light): 2 waypoints from Path 1 orbit, expanded outward
Case 5	OTAP	0.2879	23.7209	One-time	12	Path 1 (dark): 2 waypoints, same as Case 2 Path 2 (light): 2 waypoints, RAAN = 60, Inc = 60
Case 6	ISOP	0.3221	31.6395	Every orbit	24	4 waypoints, RAAN = 90, Inc = 50
Case 7	ISOP	0.3001	44.7408	Every orbit	12–18	4 waypoints, RAAN = 90, Inc = 45
Case 8	ICOT	0.3218	22.6132	One-time	24	Path 1 (dark): 2 waypoints (waypoints 2, 3 from Case 7) Path 2 (light): 2 waypoints (waypoints 1, 4 from Case 7)
Case 9	OPIT	0.2887	40.6067	One-time	12	Path 1 (dark): 2 waypoints, RAAN = 90, Inc = 45 Path 2 (light): 2 waypoints, RAAN = 90, Inc = 135

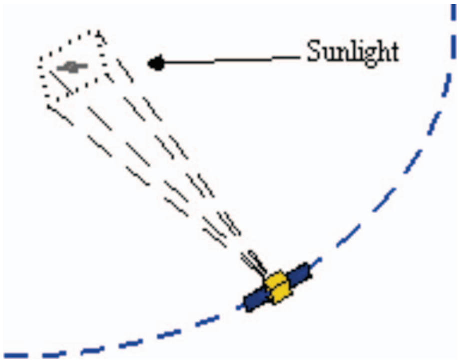

Fig. 17. Orbital geometry of chaser satellite around generic RSO: dotted lines are camera FOV.

Fig. 18. Resulting camera image at time step illustrated in Fig. 17.

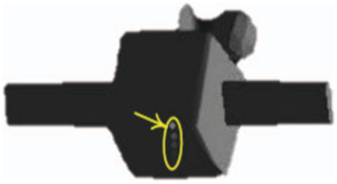

Fig. 19. Beginning of panoramic map. LRF strike points are shown inside ellipse, with first strike points indicated by arrow.

information about the RSO. The key to detecting and filling coverage gaps is the NFOV camera payload.

For simulation purposes, the apparent size and orientation of the RSO is generated from a pinhole camera model that takes camera perspective and the distance between the chaser and the RSO into account. The camera is assumed to have a 5° half-angle FOV and a 2D IR imager at the focal plane of the optics assembly that can detect the LRF blooms on the RSO. The RSO is kept within the camera FOV during the simulation. The camera and LRF are triggered at 1 Hz. The LRF is capable of triggering faster than once a second, and this constitutes an advantage to the results described in the following sections. Figs. 17 and 18 depict the simulation set-up and the resulting RSO image.

Realistic lighting conditions are implemented using the Sun as a point light source. The angle between the surface normal and the Sun vector determine shadow shading on each face of the RSO. A small level of general ambient light, 0.2 on a scale from 0 to 1, is added to account for other possible light sources such as Earth glint. For the purposes of the simulations presented here, it is assumed that the sensitivity band of the IR imager is sufficiently narrow to filter any photons emitted by the RSO.

B. Creating a Generalized Panoramic Mosaic

Panoramic imaging has been successfully used for applications such as interactive environment maps, depth mapping, object reconstruction, and smart phone applications [26–29]. Panoramas are created by iteratively merging multiple images of the same scene into one combined image. Images can be related to each other by translations, rotations, affine transformations, and arbitrary projective transformations. Because no a priori knowledge of RSO shape is assumed, it is difficult to relate positions of LRF strike points in 3D space. The panoramic image map can provide the additional information required to register and map LRF measurements. The NFOV camera takes images along with simultaneous LRF measurements: from the resulting panorama, correlations are generated of 1) the surface corresponding to each LRF strike point and 2) relative positions of strike points. Fig. 19 shows the initial iteration of the process; the trail of strike points illustrates the spatial relationship between the points.

All transformations must be calculated and applied to one or more images before the panorama can be generated. Widely used methods such as spherical projections or feature matching [30] that accomplish this assume certain restrictions on camera movement or position. The mission scenarios presented here violate these assumptions: 1) the chaser, hence the NFOV camera, is constantly moving; 2) the distance between the camera and the RSO varies; and 3) images are captured at many different angles with respect to the RSO. As a consequence most of the transformations between images will require projections that are unknown or cannot be generalized.

The algorithm used in this paper computes projections without any assumptions about camera position, orientation, or movement [31]. The method is capable of compiling images taken at arbitrary angles, resolutions, and scales without human intervention. Given a pair of 2D images of the same scene, possibly taken in two different planes, coordinates (x, y, w) are defined to be in one image plane and coordinates (x', y', w') to be in the other image plane. (x, y) are planar coordinates and w and w' are scaling factors, such that the 2D coordinates for each image are $(x/w, y/w)$ and $(x'/w', y'/w')$, respectively. Coordinate (x, y, w) in the first image can be related to (x', y', w') in the second image through a matrix transformation:

$$\begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} = \begin{pmatrix} m_0 & m_1 & m_2 \\ m_3 & m_4 & m_5 \\ m_6 & m_7 & m_8 \end{pmatrix} \begin{pmatrix} x \\ y \\ w \end{pmatrix} \tag{2}$$

In (2), $M = [m_0 \ldots m_8]$ is unknown. Elements $[m_0, m_1, m_3, m_4]$ represent a rigid, 2D rotation from the first image to the second; m_2 and m_5 are the pixel translation in the x and y directions; and m_6 and m_7 are the projective transformations in the x and y directions. m_8 is a matrix scaling factor, which is chosen such that $m_8 = 1$. Equation (2) can be rewritten as:

$$x' = \frac{m_0 x + m_1 y + m_2}{m_6 x + m_7 y + 1}$$

$$y' = \frac{m_3 x + m_4 y + m_5}{m_6 x + m_7 y + 1} \tag{3}$$

where x' and y' are the projections of (x,y) from the first image into the second. Beginning with the unit matrix as an initial estimate, the operator M is applied to every pixel coordinate (x,y) to compute its corresponding pixel (x', y') in the other image. The objective is to minimize the squared difference between the pixel intensity value $I(x, y)$ in one image and its corresponding intensity $I'(x', y')$ in the other image, over all pixel matches:

$$E = \sum \left[I'\left(x', y'\right) - I(x, y) \right]^2 = \sum e^2 \tag{4}$$

Matrix M is iteratively recomputed using the Levenberg- Marquardt algorithm to decrease the error until the difference in intensity at each pixel is minimized and the algorithm converges. The Levenberg-Marquardt algorithm requires calculating the partial derivatives of e with respect to $[m_0 \ldots m_7]$:

$$\frac{\partial e}{\partial m_k} = \frac{\partial I'}{\partial x'} \frac{\partial x'}{\partial m_k} + \frac{\partial I'}{\partial y} \frac{\partial y'}{\partial m_k} \tag{5}$$

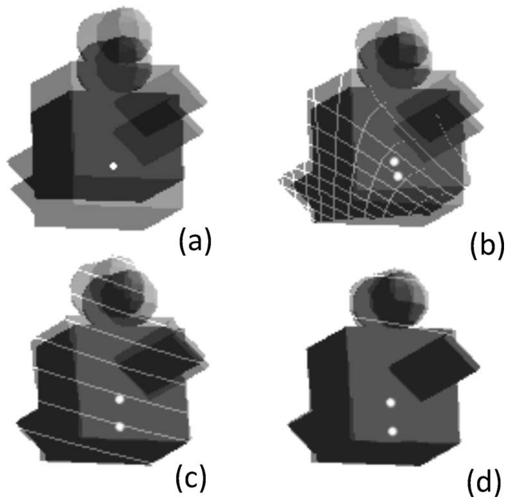

Fig. 20. Results of iterative transformations to minimize intensity difference between images: effects on LRF strike point are notable.

where $\partial I'/\partial x'$ and $\partial I'/\partial y'$ are the x and y intensity gradients of the image at (x', y'). Assuming that the chaser has a GPS receiver, each image is time-tagged with the inertial orientation and position of the camera, thus creating an accurate first estimate of the motion matrix M. The initializations create a good starting estimate for the Levenberg-Marquardt algorithm. Because the scene can be assumed to have minimal change between images, the previous transformation M_{t-1} is used to initialize the iterations for the current transformation M_t. An approximate Hessian matrix A is created using the partial derivatives in (5), where:

$$A_{kl} = \sum_{k=0}^{8} \frac{\partial e}{\partial m_k} \frac{\partial e}{\partial m_l}, \; l = 0, 1, \ldots, 8 \tag{6}$$

over all pixels. Similarly, a weighted gradient vector b is defined as:

$$b_{kl} = -\sum_{k=0}^{8} \frac{\partial e}{\partial m_k} e \tag{7}$$

Once A and b are computed for an iteration, $[m_0 \ldots m_7]$ are updated by solving the system of equations:

$$(A + I)\Delta m = b \tag{8}$$

for Δm. The elements of Δm are added to their corresponding elements in the matrix M, and the process repeats until the squared error E stops decreasing.

A typical result of the iterative process described above is shown in Fig. 20. Fig. 20a–20d are at the start, a third, two-thirds, and at the end of a successful Levenberg-Marquardt juxtaposition of two images separated by 40 s of chaser motion around the RSO. Fig. 20a shows the two images overlapped, with the LRF strike points occupying the same pixel coordinate. As the iteration progresses, and the error in M becomes smaller, the algorithm approaches the final projection of the first image into the second. Fig. 20d shows the actual location of the two strike points, 40 s apart, as they appear on the transformed RSO image. Despite the chaser's displacement and attitude slew motion, all coordinates from the later image can be represented in terms of the coordinates from the first image by applying the final M as a projection to it and overlapping both images. For further details, the reader is referred to [31].

The method is employed to build a panoramic map onboard the chaser using images taken from the NFOV camera. In an operational mission scenario, postprocessing might fall behind the rate at which images are taken. Shum and Szeliski [32] outline several heuristics to speed up postprocessing. Therefore, it is believed that the image projections can in fact be computed on-line in a reasonable amount of time, especially if the algorithm is executed in parallel. Future work includes verifying this assumption on a real-time test bed such as ARAPAIMA's Chronos [33], a desktop spacecraft simulator and isochronal payload emulator testbed used to run flight software algorithms and establish baseline payload performance in laboratory-simulated mission scenarios.

C. Minimizing Distortion: Anchor Image Projections

Because a full 360° view of the RSO is imaged multiple times, the map cannot be stored as one flat image due to the massive amount of distortion that would be introduced at the edges of the image. For rapid access to images at certain viewpoints, the map is stored as a series of anchor images, maintaining a record of the calculated projections of other images into the anchor image planes.

Anchor images can be chosen based on several factors. Computationally, it is effective to choose a certain number of images evenly distributed in time from each other in the overall image library. In the simulation presented, after one orbit of the RSO has been completed, all projections of other images into the relevant anchor images are computed. For every new image captured, its projection into the relevant anchor images is computed and stored. The stored information is then used to spatially relate LRF strike points using panoramic mapping.

It is assumed that, in most cases, a strike point is visible in the image taken simultaneously with it as a bright spot. This assumption will likely not hold if 1) the beam misses the RSO, 2) the beam is absorbed by an object, or 3) the image is taken in conditions of extreme differential brightness, such as solar glint in the optics of the NFOV camera or hot spots in the IR camera sensitivity band. In these cases, the location of the LRF strike can still be inferred based on knowledge of the camera's resolution and the physical location of the LRF instrument relative to the camera. Therefore, every time projections are mapped, the projection operation can also be applied to the bright LRF pixel coordinates of the LRF strike in the image. This yields the location of that strike point in an anchor image or any other image desired if a series of projections is used.

All of the strike points projected into the nearest anchor image can then be back-projected into the current camera image. Strike points that fall within the boundary of the RSO in the current image are considered visible and are used for detection of a gap within the current camera FOV. This observation leads to the on-line and possibly real-time generation of attitude maneuvers to fill the detected coverage gap. As the anchor images cover multiple perspectives of the RSO, especially if the RSO is tumbling, a single image may not project well into more than a few of the anchor images due to parallax error. Images that are too dissimilar result in the Levenberg-Marquadt algorithm iterating without noticeably reducing the error. Currently, if images fail to align to an anchor image within an imposed time limit, the algorithm is stopped for the respective pair of images, and it is assumed that there is no projection between them. However, if one image is projected into a second image by matrix $M1$ and a second image into a third by $M2$, the first image can be projected onto the third image by multiplying $M1$ and $M2$ to create a new projection matrix. This chaining of projections allows the relation of two images even if they failed to align with Szeliski's method as long as a series of images exists between them that are related by projections.

D. Practical Considerations to Projection Fidelity

Due to the specific requirements of the mission scenario under consideration, the application of computer vision methods raises several interesting camera-related questions. These are listed with a suggested set of corrective actions.

1) *Effect of Relative Sun Position*: Solar illumination of the RSO influences the fidelity of the mapping between regular images and anchor images. Images taken from the same viewpoint of the RSO may fail to align at all if one was taken while the visible surfaces were well lit and the other was taken in poor lighting or with different shadows present. Worse, if a badly lit image is accidentally selected as an anchor image, it could result in a large blind spot on the map where no images project well. Long periods of unusable images may result in false gaps where the algorithms fail to integrate valid strike points with the map.

This has been mitigated by exploiting the knowledge that lighting changes happen at a slower rate than the rate of image capture. A short time lapse between consecutive images provides some tolerance against lighting changes. However, onboard image exploitation will have to be suspended during LEO terminator transitions.

2) *LRF Beam Jitter*: Beam blurring, chaser attitude jitter, and differential drag can introduce noise-like components into the motion of the LRF beam. However, given the large amount of data being collected and the multiple orbits of the chaser around the RSO, it has been determined that simple linear interpolation is sufficient to place LRF strike points that occur between image captures on the map.

3) *Poor Camera Resolution and LRF Bloom Imaging*: Due to the finite resolution of the 2D planar sensors, strike points might go undetected or be

indiscernible (merged points). Another problem is the dispersion of the LRF beam, which results in a strike point becoming a so-called bloom with finite dimensions. Specifically, the projection of the bloom on a surface perpendicular to the optical axis of the LRF transmitter is a circle.

In the simulations presented here, images are scaled to approximately 12 pixels/m, i.e., LRF blooms that are less than 1/12th of a meter apart will appear in the same pixel when projected into an image. The loss of resolution is not an issue if the chaser is sufficiently far away from the RSO. However, this could quickly become a problem if the chaser is tasked to close the distance to the RSO and bases its motion on lower-resolution relations between strike points. Therefore, camera resolution was chosen during the ARAPAIMA preliminary design phase to balance against the closest approach distance desired for mission execution.

VII. COVERAGE GAP DETECTION WITH VORONOI DIAGRAMS

A. Voronoi Diagrams

Also known as Dirichlet tessellation, a Voronoi diagram is a method of decomposing a 3D space based on the distance to a specified subset in that space. In mathematics, this subset is referred to as a family of sites or generators. Each generator is associated with a corresponding Voronoi cell. In 3D Euclidean space, this corresponds to the set of all points in that space whose distance to the given generator is less than the distance to every other generator [34]. This technique allows for a multidimensional space to be uniformly partitioned into subspaces and analyzed using computational geometry, despite large data sets [35]. Voronoi diagrams have found a number of uses in applications such as detecting uncovered areas within a sensor network [36], clustering data sets, and surface reconstruction from point clouds.

Given a data set D, all of the data points p in D are said to be the centers of Voronoi cells. The Voronoi cell of p is defined as the region that consists of all points that are closer to p than to any other cell center in D according to a Euclidean distance metric. The collection of tessellated Voronoi cells makes up the Voronoi diagram for the given data set (Fig. 21a). The edges between two cells indicate that all points on that line are equidistant from the respective cell centers.

The data set used by the method described here consists of a collection of LRF strike points imposed onto a panoramic 2D NFOV camera image where the centers of the Voronoi cells are the coordinates of the strike points. The geometry of the resulting Voronoi diagrams is used to examine the areas in the NFOV image where multiple cells meet. The intersection of two or more edges between cells is designated as a Voronoi point. The definition forms the fundamental concept for efficient coverage gap detection: a Voronoi point is the same distance to all strike points whose cells touch it. The distance is the radius of a circle that passes through those strike points. The circle is called an empty circle, as shown in Fig. 21b. Therefore, a gap in LRF strike point coverage can be found in a given panoramic image containing all visible strike points by computing its Voronoi diagram and finding the largest empty circle over all Voronoi points. The largest circle can be related to positions recorded on the map created with the panoramic mapping methods previously described.

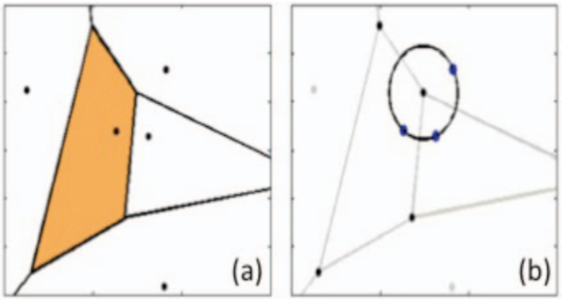

Fig. 21. Voronoi diagrams constructed for $p = 5$ data points. Complete Voronoi circle is highlighted in (a), and (b) shows empty circle for one Voronoi point.

B. Overview of the Gap Detection Method

During the first phase of the simulation scenario, the chaser orbits the RSO under a preprogrammed scouting maneuver profile similar to those discussed in Section IV. This occurs for a predetermined time on the order of a few orbits, until there are sufficient images to generate a panoramic map of the RSO. Once the map is created, a list of visible strike points and their positions are computed for any arbitrary view of the RSO using image projections. The positions of the LRF strike points are recorded on the map in pixel coordinates.

The chaser then enters the second phase during which it performs active gap filling using current camera images to determine what region on the map it is seeing. This allows the algorithm to dynamically calculate the position of the gap in its FOV, determine the slew angle needed to move the beam toward it, and generate the commands to slew. Because the slew angle is recalculated at each time step, the spacecraft corrects itself as the LRF beam moves closer to the center of the gap. The process is repeated after the detected gap passes out of range and the algorithm targets a new gap.

VIII. ON-LINE GAP DETECTION AND FILLING

The procedure for coverage gap detection is easily automated by combining path navigation (Section III), panoramic mapping (Section VI), and gap detection via Voronoi diagrams (Section VII). Fig. 22 illustrates the algorithm.

Once the position of the RSO is identified in the integrated panoramic image, given knowledge of the image's position in the panoramic map, the algorithm decides which strike points are visible and projects them into the most current image. A series of Voronoi diagrams

Fig. 22. Gap detection algorithm.

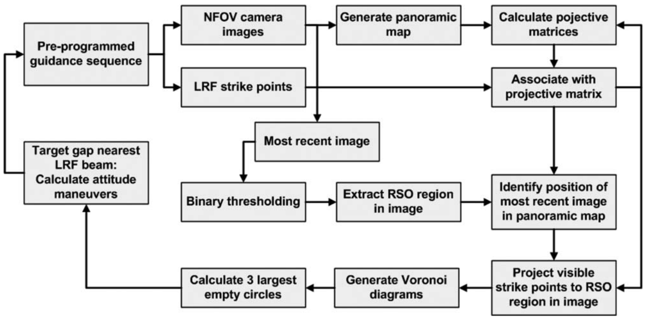

Fig. 23. Largest empty circles as mapped to LRF coordinates. Unmarked gap on RSO lower half appears larger than marked gaps due to image perspective.

are generated using the strike points as Voronoi cell centers. For each Voronoi point within the RSO region, the distance between it and its nearest strike point is calculated. Fig. 23 shows the three largest empty circles marked in this manner so that the gap nearest the current position of the beam can be easily targeted for chaser attitude commanding. This avoids a slew that sends the beam too far away overly quickly. The gaps generated are related spatially to each other and to the RSO by the map. Coordinates of the coverage gap can then be recorded by projecting it into the anchor image nearest to the current image. These coordinates are queried and the target gap is chosen as the one nearest to the beam location, measured in pixels.

Next, attitude maneuvers to fill the gap can be generated on-board the spacecraft in real time. The angle between the current beam position and the gap center is calculated with:

$$\theta = arctan\frac{p}{d} \tag{9}$$

where p is the distance from the beam position in the image to the gap center and d is the distance from the spacecraft to the RSO (known by the LRF measurement), both measured in pixels:

$$d = \frac{h}{2m\tan(0.5v)} \tag{10}$$

where h is the image height in pixels, m is the LRF measurement in meters, and v is the camera FOV in radians along the camera y-axis. If the beam has missed the RSO, there is no LRF measurement value and the image is discarded. The angle θ is then used to slew from the current position to reach the gap and pulse the LRF to gather point cloud information.

IX. RESULTS

The gap detection and filling algorithm was run in conjunction with the maneuver profiles described in Section IV against simulated images similar to those in Fig. 20. From Table I, test cases 3–5 and 9 are the best scenarios, which reach the goal of an edge metric score of 0.3 in 12 h of surveillance. Test case 6 is a moderate scenario and can reach the same goal in 24 h of surveillance. Test case 2 is an example of an inefficient scenario and fails to meet this goal after 36 h of surveillance.

Each test case is first baselined with no gap detection for 12 and 18 h of surveillance time. Next, each test case is run for the same amount of time, with 2 h of general slew maneuvers replaced by 2 h of gap detection maneuvers. The time is divided as follows: for the 12 and 18 h surveillance time, respectively, the chaser executes general slew maneuvers (striping along either axis and spiraling), for 5 or 8 h, then runs gap detection for 1 h, then repeats these cycles. The reason for this separation is linked to the command profiles—some cases perform a propulsive maneuver onto another path halfway through surveillance. The separation allows for gap detection to be performed on both paths equally.

The ratio between general slewing maneuvers and closed loop gap detection slewing maneuvers is not intuitive. Performing gap detection for more than half of the total surveillance time results in patches of extremely dense LRF point coverage, but coverage over the RSO and overall edge metric score are worse than that exhibited by general slew maneuvers only. It is concluded that if improvement in overall density is desired, then time spent performing gap detection should be small relative to the total amount of surveillance time. Longer periods of on-line gap detection are recommended to focus on a small area, such as a particular instrument in the sensor pod on the RSO or when scouting an area planned for docking or on-orbit repair operations.

Figs. 24 and 25 show the percentage of total improvement against the general slewing cases for 12 and 18 h of total surveillance, respectively. With just 2 h of gap detection time, the result of incorporating the methods described in this paper is a significant improvement in all cases, whether labeled as good, moderate, or bad.

X. SUMMARY AND CONCLUSIONS

A method has been developed that combines various relative orbits of a chaser satellite about a target RSO with various scanning attitude profiles to generate 3D point clouds with a single-beam LRF. An edge detection algorithm has been employed to extract information from the point cloud and assign a fitness metric to the point clouds against a ground truth model of the RSO. It has

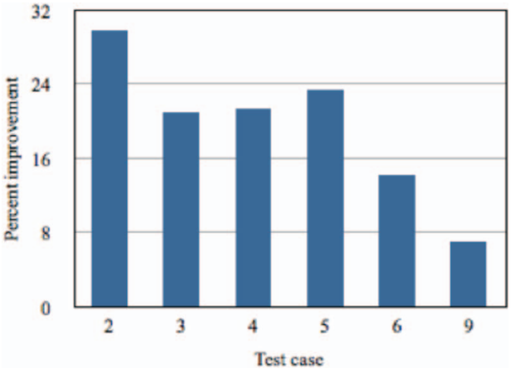

Fig. 24. Percentage of improvement against general slewing case: after 12 h of surveillance.

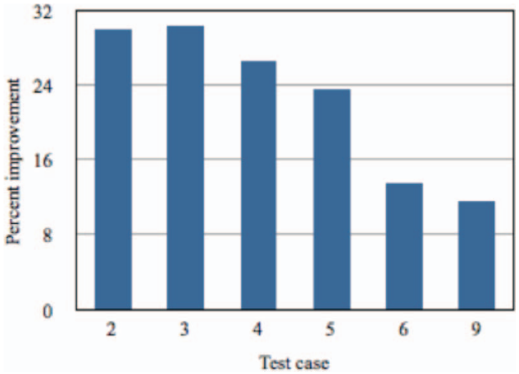

Fig. 25. Percentage of improvement against general slewing case: after 18 h of surveillance.

been found that the edge metric score levels off logarithmically even for the best test cases. This suggests that maneuvers that change the orientation of the relative orbit of the chaser with respect to the fundamental plane of the target are more important than increasing the surveillance time. Five types of relative motion profiles are combined with three types of attitude motion for a detailed analysis. Results of the simulation show that starting with a wide slew angle and changing to a smaller angle after a certain period of time ensures both widespread coverage and high resolution. Out-of-plane transfers are most time efficient because they fill up the point clouds fastest. The maneuvers can be designed to have a low ΔV cost if the transfer between orbit paths is planned carefully.

To further improve on these profiles, an on-line method to autonomously fill the gaps in point cloud coverage is presented. Adding this algorithm to the pregenerated command profiles results in improvement for all the maneuver profiles. Percentage improvements in point cloud fitness scores range from 7%–31%. Under baseline conditions, test case 2 fails to reach a score of 0.3 after over 36 h of surveillance. With gap detection, the score is below 0.3 after just 12 h. This exhibits a clear and significant improvement to previous methods in reconstruction of the RSO and space situational awareness (SSA) objectives.

Most importantly, both cases 2 and 6 have orbital paths in one plane only. However, the gap-filling method described here produces scores and point cloud sizes nearly as good as the cases that have two distinct planes of orbital motion. This suggests that using on-line attitude commanding with gap detection methods can reduce the need for propulsive maneuvers and reduce overall surveillance time. Future work will apply the methods described here to optimize the relative maneuver and attitude command profiles as well as improving the fidelity of gap detection with better filtering methods for background objects. For example, attitude commanding can be implemented with a potential function in which regions of low coverage are the valleys of the potential field. The relative translational motion between the chaser and the RSO will aid the cases in which the attitude motion is trapped in local minima. Additional improvements can be implemented in the form of attitude commanding similar to the saccades of the human eyes to cover certain regions of the RSO.

The results of the work presented here validate the utility of a single-beam LRF as a tool for 3D imaging of RSO shapes. These methods are directly applicable to power-constrained space-based SSA and proximity operations missions.

ACKNOWLEDGMENT

The authors would also like to thank two anonymous reviewers whose comments helped strengthen this manuscript.

REFERENCES

[1] Zimpfer, D., Kachmar, P., and Tuohy, S.
Autonomous rendezvous, capture and in-space assembly: Past, present and future.
In *Proceedings 1st Space Exploration Conference: Continuing the Voyagage of Discovery*, AIAA 2005–2523, 2005, 1–12.

[2] Woffinden, D. C., and Geller, D. K.
Navigating the road to autonomous orbital rendezvous.
Journal of Spacecraft and Rockets, **44**, 4 (Jul. 2007), 898–909.

[3] Tatsch, A., Fitz-Coy, N., and Gladun, S.
On-orbit servicing: A brief survey.
Presented at the *Proceedings of the Performance Metrics Intelligent Systems Conference*, 2006.

[4] Goodman, J. L.
History of space shuttle rendezvous and proximity operations.
J ournal of Spacecraft and Rockets, **43**, 5 (Sep. 2006), 944–959.

[5] Lauretta, D.
An overview of the OSIRIS-REx asteroid sample return mission.
In *Proceedings 43rd Lunar and Planetary Scientific Conference*, The Woodlands, TX, **LPSC** (2012), 5–6.

[6] Berry, K., Sutter, B., May, A. and Williams, K.
OSIRIS-REx touch-and-go (TAG) mission design and analysis.
Presented at the *Proceedings 36th Annual AAS Guidance and Control Conference*, AAS 13–095, 2013.

[7] Smith, P. H., Rizk, B., Kinney-Spano, E., and Merrill, C.
The OSIRIS-REx camera suite (OCAMS).
In *44th Lunar Planetary Scientific Conference*, 2013–1690, 2013, 1–2.

[8] Udrea, B., Nayak, M. V., and Ankersen, F.
Analysis of the pointing accuracy of a 6U CubeSat mission for proximity operations and resident space object imaging.
Presented at the *Proceedings of the 5th International Conference on Spacecraft Formation Flying Missions and Technologies*, 2013.

[9] Udrea, B., Nayak, M. V., Ryle, M., Martini, N., Gillespie, S., Grande, T., Caicedo, S., Baba, A., Harris, K., DiGregorio, J., Patel, P., and Huang, A.
Mission design and concept of operations of a 6U CubeSat mission for proximity operations and RSO imaging.
In *Proceedings of the 5th International Conference on Spacecraft Formation Flying Missions and Technologies*, 2013, 1–15.

[10] Franquiz, F., Edwards, P., and Udrea, B.
Attitude determination and control system design for a 6U CubeSat for proximity operations and rendezvous.
In *Proceedings AAS/AIAA Astrodynamics Specialist Conference*, 2014, 1–18.

[11] Samson, C., English, C., Deslauriers, A., Blais, F., and Ferrie, F.
Neptec 3-D laser camera system: From space mission STS-105 to terrestrial applications.
Canadian Aeronautics and Space Journal, **50**, 2 (2004), 115–123.

[12] Taati, B., Bondy, M., Jasiobedzki, P., and Greenspan, M.
Automatic registration for model building using variable dimensional local shape descriptors.
In *Proceedings 6th International Conference 3-D Digital Imaging Modeling*, 2007, 1–8.

[13] English, C., Zhu, S., Smith, C., and Ruel, S.
TriDAR: A hybrid sensor for exploiting the complementary nature of triangulation and LIDAR technologies.
Presented at the *Proceedings 8th i-SAIRAS*, 2005.

[14] Naasz, B. J., Berry, M. M., Kim, H., and Halls, C. D.
Integrated orbit and attitude control for a nanosatellite with power constraints.
In *Proceedings AAS/AIAA Space Flight Mechanics Conference*, **AAS 03–100**, 540 (2003), 1–18.

[15] Harris, K., Ryle, M., Martini, N., and Patel, P.
Application for RSO Automated Proximity Analysis and IMAging (ARAPAIMA): Development of a NanoSat-based space situational awareness mission.
In *Proceedings of the Small Satellite Conference*, 2013, 1–19.

[16] Hevner, R., Holemans, W., Puig-Suari, J., and Twiggs, R.
An advanced standard for CubeSats.
In *Proceedings Small Satellite Conference*, SSC11–II–3, 2011, 1–9.

[17] Le, B. Q., Cole, T. D., Rodriguez, D. B., Reiter, R. A., Moore, R. C., Boies, M. T., and Schaefer, E. D.
NEAR laser rangefinder light-weight packaging design.
In *Proceedings SPIE 2811, Photonics Space Environment IV*, **2811** (1996), 208–216.

[18] Nayak, M. V., Beck, J., and Udrea, B.
Design of relative motion and attitude profiles for three-dimensional resident space object imaging with a laser rangefinder.
In *Aerospace Conference 2013 IEEE*, 2013, 1–16.

[19] Nayak, M. V., Beck, J., and Udrea, B.
Real-time attitude commanding to detect coverage gaps and generate high resolution point clouds for RSO shape characterization with a laser rangefinder.
In *Aerospace Conference 2013 IEEE*, 2013, 1–18.

[20] Harris, K., Baba, A., DiGregorio, J., Nayak, M. V.m and Udrea, B.
Experimental characterization of a miniature laser rangefinder for resident space object imaging.
In *23rd AAS/AIAA Spaceflight Mechanics Conference Proceedings*, AAS 13–480, 2013, 1–12.

[21] Nayak, M. V., Udrea, B., Marsella, B., and Beck, J.
Application of a laser rangefinder for space object imaging and shape reconstruction.
AAS Spaceflight Mechanics Conference, AAS 13–429 (2013), 1–21.

[22] Pauly, M., Gross, M., and Kobbelt, L.
Efficient simplification of point-sampled surfaces.
In *Proceedings of the IEEE Computer Society Conference on Visualization*, **4** (2002), 163–170.

[23] Gurfil, P., and Kasdin, N. J.
Nonlinear modeling of spacecraft.
Journal of Guidance Control and Dynamics, **27**, 1 (2003), 154–157.

[24] Schaub, H.
Relative orbit geometry through classical orbit element differences.
Journal of Guidance Control and Dynamics, **27**, 5 (2004), 839–848.

[25] Healy, L. M. and Henshaw, C. G.
Trajectory guidance using periodic relative orbital motion.
Journal of Guidance Control and Dynamics, to be published.

[26] Brown, M., and Lowe, D.
Recognizing panoramas.
In *Proceedings Ninth IEEE International Conference on Computer Vision*, **2**(2003), 1218–1225.

[27] Szeliski, R.
Image mosaicing for tele-reality applications.
In *Proceedings of the Second IEEE Workshop on Applications of Computer Vision, 1994*, **2**(1994), 44–53.

[28] Luhmann, T., and Tecklenburg, W.
3-D object reconstruction from multiple-station panorama imagery.
In *Proceedings ISPRS Workshop on Panoramic Photogrammetry*, **34**, 5 (2004), 1–8.

[29] Gledhill, D., Tian, G. Y., Taylor, D., and Clarke, D.
Panoramic imaging—A review.
Computer & Graphics, **27**, 3 (Jun. 2003), 435–445.

[30] Szeliski, R., and Shum, H.
Creating full view panoramic image mosaics and environment maps.
In *Proceedings 24th Annual Conference on Computer Graphics and Interactive Techniques*, 1997, 251–258.

[31] Szeliski, R.
Video mosaics for virtual environments.
Journal of Computer Graphics and Applications IEEE, **16**, 2 (1996), 22–30.

[32] Shum, H.-Y., and Szeliski, R.
Construction of panoramic image mosaics with global and local alignment.
International Journal of Computer Vision, **36**, 2 (2000), 101–130.

[33] Harris, K. K., Cokley, J. M., Holden, S. R., Udrea, B., Stebler, S. T., Williams, B. E., McGarvey, M. R., and Nayak, M. V.
Small-scale payload operations simulator for proximity operations.
In *AIAA Space 2014 Conference and Exposition*, **2014–4289**, 1–20.

[34] Okabe, A., Boots, B., Sugihara, K., and Chiu, S. N.
Spatial Tessellations—Concepts and Applications of Voronoi Diagrams, 2nd ed. New York, NY: John Wiley and Sons, Inc, 2000, 408–696.

[35] Aurenhammer, F.
Voronoi diagrams—A survey of a fundamental geometric data structure.
Journal of ACM Computing Surveys, **23**, 3 (1991), 345–405.

[36] Ghosh, A.
Estimating coverage holes and enhancing coverage in mixed sensor networks.
In *Proceedings 29th Annual IEEE International Conference on Local Computer Networks*, 2004, 68–76.

Michael (Mikey) Nayak received his B.S and M.S. degrees in aerospace engineering from Embry-Riddle Aeronautical University, FL. He is currently pursuing a Ph.D. degree in Earth and planetary science at the University of California, Santa Cruz. Mikey has served as lead flight director for the Air Force Research Laboratory's Tactical Satellite-III and has managed operations for the NASA Earth Science pathfinder CloudSat. He is the recipient of a National Defense Science and Engineering Graduate (NDSEG) Fellowship and the 2013 General Samuel C. Phillips award for Space and Missile System Center Outstanding Young Engineer/Scientist.

Jaclyn Beck received a B.S. degree in computer science from Northern Michigan University and an M.S. degree in computer engineering from the University of Wisconsin, Madison. She is currently a computer scientist with the Applied Engineering and Technology Directorate, NASA Goddard Space Flight Center, working on data systems for remote sensing instruments. Her research interests include machine learning and computer vision algorithm development.

Bogdan Udrea is an associate professor, Department of Aerospace Engineering, Embry-Riddle Aeronautical University, and ARAPAIMA principal investigator. Bogdan previously worked as a control and navigation systems engineer at the European Space Agency Space Research and Technology Centre (ESTEC). His research interests include spacecraft control and space mission design with emphasis on proximity operations, formation flying, near-Earth object exploration, and on-board autonomy.