Analysis on Debris Detection and Trajectory Predictions*

1st Yvette Espinoza
Software Engineer
Northrop Grumman
Redondo Beach, CA, USA
yespinoz@purdue.edu

***Abstract*—Technological improvements over the past few decades have made space exploration more feasible, but the increase in space activity has also resulted in undesirable space debris that can pose a threat to the safety of space crafts. Space surveillance systems can detect and track large debris, providing space situational awareness that reduces the risk of equipment colliding with debris, but such systems often fail to properly identify smaller debris. The main challenge lies with the uncertainty regarding the debris characteristics, making it difficult to accurately predict the orbit. Current detection and tracking of small debris is performed by either using ground based systems, or small satellite constellations to gather information on the debris' characteristics and generate orbit predictions. This research project will first look at the differences between the two systems, focusing on why small satellite constellations are preferred when considering collision avoidance and active debris removal systems. The iterative closest point algorithm will be discussed as a possible solution to pose estimation, followed by a simulation on how to algorithm works.**

I. Literature Review

Space exploration, first started in the late 1950's with the USSR's launch of Sputnik I, has become a vital part in current technologies. Civilians and government alike have come to rely on space systems, such as GNSS or weather satellites, but the addition of new satellites in an already crowded area is increasing the risk of collisions. This idea was first introduced in 1978 by Kessler, who explained the high density of objects in low Earth orbit (LEO) increases the risk of collisions, and collisions themselves would create more debris that would further increase the collision risk, a scenario commonly referred to as the Kessler Syndrome. This scenario was brought back to attention in 2009 with the first recorded collision between between two satellites, Iridium 33 and Cosmos 2251, which at the time produced 1,632 space debris fragments larger than 10 cm [17]. These fragments are large enough to be tracked by surveillance systems, such as the Department of Defense's Space Surveillance Network (SSN), but smaller fragments, referred to as small space debris, are more difficult to detect and track.

Surveillance networks are catalogs that contain characteristics of registered objects, such as the orbital parameters, which are essential for providing space situational awareness (SSA) [8]. They rely on both ground based radars and optical telescopes to track the position of the object over time, information that is needed to determine and predict orbits. However, applications such as debris collision warning, or active debris removal require information on smaller debris, along with a higher accuracy in orbit determination and prediction, both of which SSN's struggle to provide [6].

A. Small Satellite Constellations

In an effort to increase the accuracy of space debris orbit detection and predictions, studies were conducted to validate the concept of space based space surveillance. In 1996 the Midcourse Space Experiment satellite was launched with a Space-Based Visible (SBV) sensor, becoming the first demonstration of space-based surveillance that could be valuable to the SSN as an addition to the ground based systems [15]. In the following years more satellites, such as the SBSS Block 10 and the Sapphire, have been deployed and shown to provide a more accurate description of the environment [3]. Onboard sensors have an unobstructed view of debris, not affected by the weather as optical systems are, and can provide more accurate measurements when processing occurs onboard the spacecraft instead of relying on ground control centers.

B. Pose Acquisition and Tracking

One method for onboard pose determination and tracking of space debris involves using a model-based algorithm that processes 3-D point clouds collected from a LIDAR sensor [11][7][10]. The first step is to obtain an initial pose, then refine the pose measurement as part of the tracking and orbit estimation. For the initial pose $p_0 = (\alpha_0, \beta_0, \gamma_0, T_0)$, [11] proposed a three step method to estimate the relative position vector directly from the measured point cloud, referred to as 3-D PCA-based online Template Matching.

First, the initial position vector (T_0) of the satellite with respect to the target, or debris to track, is computed as the centroid of the point cloud. Then the main axes (e_m) are estimated using the Principle Component Analysis, where the directions are given by the eigenvectors of the data's covariance matrix (Q). The main axes equations are shown in (1), while the roll (α_0) and pitch (β_0) estimates are in (2).

$$\begin{bmatrix} e_{M_x} \\ e_{M_y} \\ e_{M_z} \end{bmatrix} = \begin{bmatrix} cos(\beta_0) & 0 & -sin(\beta_0) \\ sin(\alpha_0)sin(\beta_0) & cos(\alpha_0) & sin(\alpha_0)cos(\beta_0) \\ cos(\alpha_0)sin(\beta_0) & -sin(\alpha_0) & cos(\alpha_0)cos(\beta_0) \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{1}$$

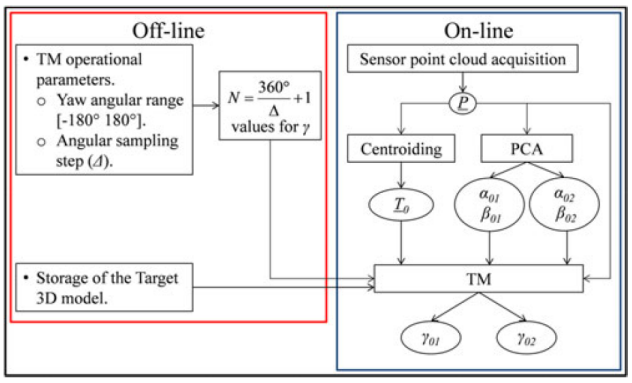

Fig. 1. 3-D PCA-based online TM algorithm.

$$\begin{cases} \alpha_0 = tan^{-1}(\frac{e_{M_x}}{e_{M_z}}) \\ \beta_0 = sin^{-1}(-e_M x) \end{cases} \tag{2}$$

An overview of the model from acquisition of the point cloud data to the inital pose estimation, taken from [11] is shown in Figure 1.

II. Simulation Environment

The simulated environment, which takes a known 3-D model and simulates LIDAR point cloud data that would be collected from an orbiting satellite, was generated in accordance with what was done in [7].

A. Coordinate System and Pose Parameters

Liu, et al [7], defined four reference frames, shown in Figure 2. The left side shows the chaser, or small satellite O_c, which would be equipped with a LIDAR sensor O_s. The right side shows the target model O_m, which could be any debris of interest, and the model as detected by the LIDAR sensor, O_t.

The target model O_m is a point cloud representation of a known 3-D model that is used to estimate the transformation from O_s to O_m. In this transformation a point from the target model, $P_m(x_m, y_m, z_m)$ is transformed into a matching point in the sensor frame $P_s(x_s, y_s, z_s)$ through equations (3) - (8), where ϕ is the rotation about the X-axis, θ the rotation about the Y-axis, and γ the rotation about the Z-axis.

$$T = \begin{bmatrix} \Delta x & \Delta y & \Delta z \end{bmatrix}^T \tag{3}$$

$$R_X(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos(\phi) & sin(\phi) \\ 0 & -sin(\phi) & cos(\phi) \end{bmatrix} \tag{4}$$

$$R_Y(\theta) = \begin{bmatrix} cos(\theta) & 0 & -sin(\theta) \\ 0 & 1 & 0 \\ sin(\theta) & 0 & cos(\theta) \end{bmatrix} \tag{5}$$

$$R_Z(\gamma) = \begin{bmatrix} cos(\gamma) & sin(\gamma) & 0 \\ -sin(\gamma) & cos(\gamma) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{6}$$

$$R = R_Y(\theta) \times R_X(\phi) \times R_Z(\gamma) \tag{7}$$

$$P_s = RP_m + T \tag{8}$$

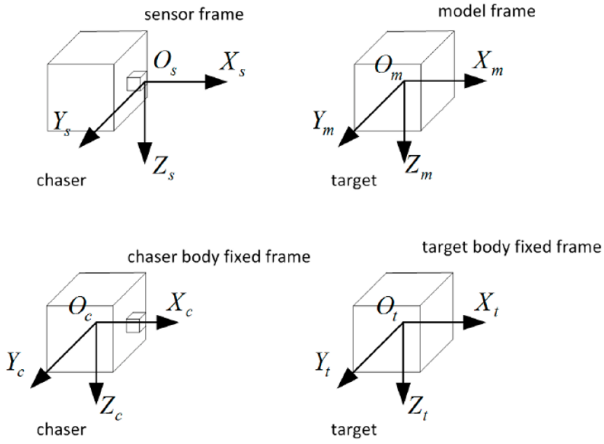

Fig. 2. Reference frames definition.

B. Target Model

Envisat, launched by the European Space Agency in 2002, was initially selected as the target model because of its large size and relatively simple shape. The satellite, which is 26 m (85 ft) × 10 m (33 ft) × 5 m (16 ft) in orbit, was decommissioned and marked as debris after it lost communication in 2012 [2]. A 3-D model was obtained from [16], then converted to point cloud data. The model, shown in Figure 3, was not to scale, but since the goal of the project was to implement the pose estimation algorithms, it was deemed suitable for the task.

C. Simulated LIDAR Data

The SR4000 camera from MESA Imaging was selected as the sensor from which the data is simulated. The specifications were obtained from [5] and are shown in Table I.

TABLE I
SR4000 Specifications

Sensor pixels	176 x 148
Frame rate	54 (f/s)
Focus Length (f)	10 (mm)
Scan resolution at 3 m	13.6 (mm)
Field of view (α_h, α_v)	43.6 x 34.6 (degrees)

The first step in generating the simulated data is to take the target model point cloud and use equations (3) - (8) to get the sensor's view, based on the input rotation angles (ϕ, θ, γ), and observed position (x, y, z).

To accurately simulate whether the points line in the sensors field of view the following equations are used

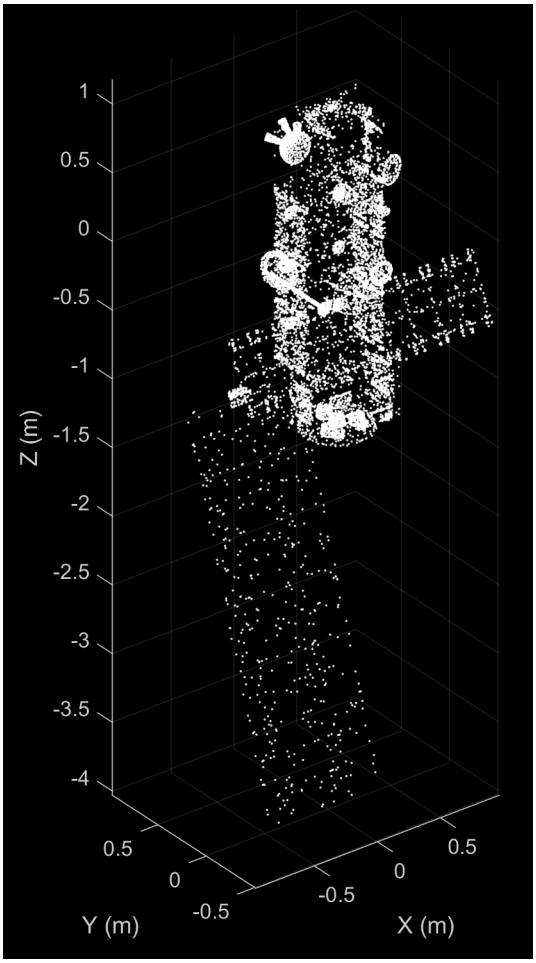

Fig. 3. ENVISAT 3-D Point Cloud Model.

$$\begin{cases} Y_{range} = \begin{bmatrix} -Dtan(\alpha_h/2), & Dtan(\alpha_h/2) \end{bmatrix} \\ Z_{range} = \begin{bmatrix} -Dtan(\alpha_v/2), & Dtan(\alpha_v/2) \end{bmatrix} \end{cases} \tag{9}$$

where Y_{range} and Z_{range} are the horizontal and vertical observable ranges, D is the x component of P_s. The horizontal and vertical view angles (α_h, α_v) are provided in Table I. If the calculated sensor point P_s is outside of the sensor range, the point is discarded. To account for distance errors, a random value in the range $[-\Delta d, \Delta d]$ was also added.

III. Results

Matlab was used to take the Envisat 3-D model and convert it to a point cloud file, and Python was used to generate the simulated LIDAR data. The Liu, et al. paper ([7]) was followed when generating the data, but some assumptions were made when adding the distance measurement errors. The SR4000 camera has a measurement accuracy of 1 cm, but the paper did not discuss how they added their error. The error was assumed to have a Gaussian distribution, zero mean and 1 cm standard deviation, with the error added randomly to one of the three axis.

The code for the project can be found here: https://github.com/Yvette4/AAE575

A. Experiment 1

The first simulated data was for an observed position (x,y,z) at (10, 0, 0) meters and initial rotation angles (ϕ, θ, γ) at (-180, 0, 0) degrees. The generated cloud point is shown in Figure 4.

[NOTE: THE FINAL VERSION OF THE PAPER WILL ROTATE THE OBSERVABLE ANGLE γ BY 10 DEGREES TO GENERATE MORE POINT CLOUDS AND APPLY THE EQUATIONS LISTED IN THE LITERATURE REVIEW TO GET ESTIMATED ANGLES AND POSITIONS. THESE WILL BE COMPARED WITH THE REAL ANGLE AND POSITION TO SEE HOW EFFECTIVE THE ALGORITHM IS]

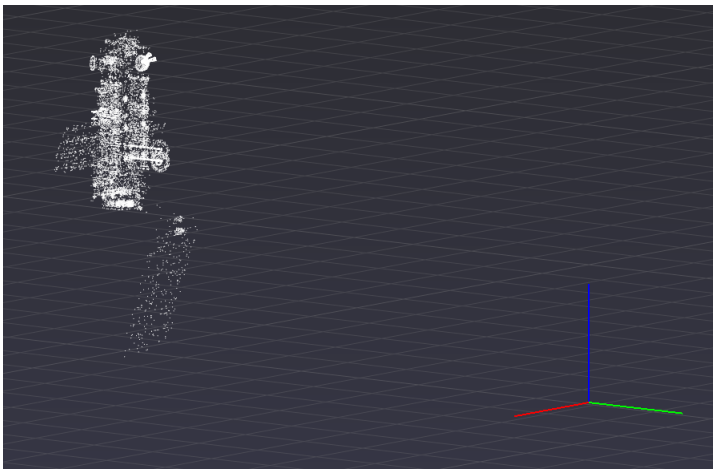

Fig. 4. Generated point cloud at 10 meters view distance.

B. Experiment 2

The next observed position was (2, 0, 0) meters and rotation angles (-180, 0, 0) degrees. The generated cloud point is shown in Figure 5. Since the target was at a close range, the entire model did not fit in the sensor's field of view, which did not detect the bottom panel.

References

[1] James Bennett et al. "Accurate orbit predictions for debris orbit manoeuvre using ground-based lasers". In: *Advances in Space Research* 52 (Dec. 2013), pp. 1876–1887. DOI: 10.1016/j.asr.2013.08.029.

[2] *Envisat Overview*. URL: https://earth.esa.int/eogateway/missions/envisat/description.

[3] Leonard Felicetti and M. Reza Emami. "A multi-spacecraft formation approach to space debris surveillance". In: *Acta Astronautica* 127 (2016), pp. 491–504. ISSN: 0094-5765. DOI: https://doi.org/10.1016/j.actaastro.2016.05.040. URL: https://www.sciencedirect.com/science/article/pii/S0094576516301060.

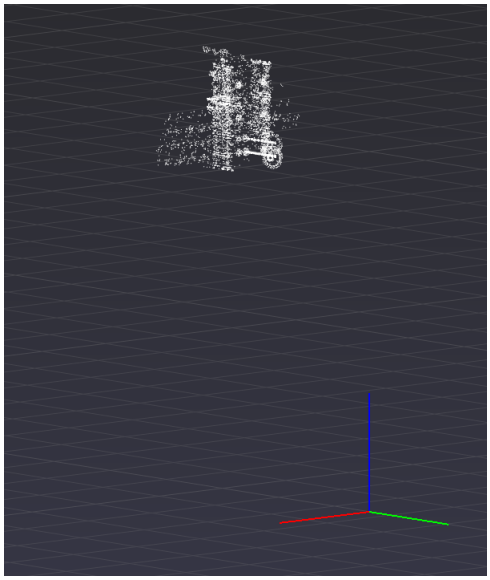

Fig. 5. Generated point cloud at 2 meters view distance.

[4] Simon Kim et al. "Analysis of Space Debris Orbit Prediction Using Angle and Laser Ranging Data from Two Tracking Sites under Limited Observation Environment". In: *Sensors* 20.7 (2020). ISSN: 1424-8220. DOI: 10.3390/s20071950. URL: https://www.mdpi.com/1424-8220/20/7/1950.

[5] Tobias K. Kohoutek, Rainer Mautz, and Jan D. Wegner. "Fusion of Building Information and Range Imaging for Autonomous Location Estimation in Indoor Environments". In: *Sensors* 13.2 (2013), pp. 2430–2446. ISSN: 1424-8220. DOI: 10.3390/s130202430. URL: https://www.mdpi.com/1424-8220/13/2/2430.

[6] Bin Li et al. "A Machine Learning-Based Approach for Improved Orbit Predictions of LEO Space Debris With Sparse Tracking Data From a Single Station". In: *IEEE Transactions on Aerospace and Electronic Systems* 56.6 (2020), pp. 4253–4268. DOI: 10.1109/TAES.2020.2989067.

[7] Lujiang Liu, Gaopeng Zhao, and Yuming Bo. "Point cloud based relative pose estimation of a satellite in close range". In: *Sensors* 16.6 (2016), p. 824.

[8] Jaime Gerson Cuba Mamani and Pablo Raúl Yanyachi Aco Cardenas. "LiDAR small satellite for space debris location and attitude determination". In: *2019 IEEE International Conference on Aerospace and Signals (INCAS)*. 2019, pp. 1–6. DOI: 10.1109/INCAS.2019.8827211.

[9] Alessia Nocerino et al. "LIDAR-based multi-step approach for relative state and inertia parameters determination of an uncooperative target". eng. In: *Acta astronautica* 181 (2021), pp. 662–678. ISSN: 0094-5765.

[10] Roberto Opromolla and Alessia Nocerino. "Uncooperative Spacecraft Relative Navigation With LIDAR-Based Unscented Kalman Filter". In: *IEEE Access* 7 (2019), pp. 180012–180026. DOI: 10.1109/ACCESS.2019.2959438.

[11] Roberto Opromolla et al. "Pose Estimation for Spacecraft Relative Navigation Using Model-Based Algorithms". In: *IEEE Transactions on Aerospace and Electronic Systems* 53.1 (2017), pp. 431–447. DOI: 10.1109/TAES.2017.2650785.

[12] Roberto Opromolla et al. "Uncooperative pose estimation with a LIDAR-based system". eng. In: *Acta astronautica* 110 (2015), pp. 287–297. ISSN: 0094-5765.

[13] Shanmugam Raguraman, Raj N.S Sarath, and Jerrin Varghese. "Space Debris Removal: Challenges and Techniques-A Review". In: *2020 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*. 2020, pp. 1361–1366. DOI: 10.1109/ICRITO48877.2020.9197877.

[14] Jizhang Sang, James Bennett, and Craig Smith. "Experimental results of debris orbit predictions using sparse tracking data from Mt Stromlo". In: *Acta Astronautica* 102 (Sept. 2014), pp. 258–268. DOI: 10.1016/j.actaastro.2014.06.012.

[15] Grant Stokes et al. "The space-based visible program". In: *Space 2000 Conference and Exposition*. 2000, p. 5334.

[16] tashtegoFollow. *Envisat - 3D model by Tashtego (@tashtego)*. URL: https://sketchfab.com/3d-models/envisat-65b0ec49681a44f68dfc8bd4efe95839.

[17] Ting Wang. "Analysis of Debris from the Collision of the Cosmos 2251 and the Iridium 33 Satellites". In: *Science & Global Security* 18 (2010), pp. 118–87.

[18] David Zuehlke et al. "An End-to-End Process for Local Space Situational Awareness from Optical Observers". In: *2020 IEEE/ION Position, Location and Navigation Symposium (PLANS)*. 2020, pp. 1547–1555. DOI: 10.1109/PLANS46316.2020.9109868.