Deep learning-based spacecraft pose estimation for the pre-capture phase scenario

Roman Prokazov[1,a*]

[1]Department of Industrial Engineering, Alma Mater Studiorum Università di Bologna Via Fontanelle 40, 47121, Forlì, Italy

[a]roman.prokazov2@unibo.it

Keywords: Deep Learning, Computer Vision, Spacecraft Pose Estimation, Synthetic Data

Abstract. The increasing population of space debris in Low-Earth Orbit (LEO) poses a significant threat to operational satellites and future space endeavors. To address this challenge, leading aerospace companies worldwide are developing on-orbit servicing and debris removal satellites. These servicer satellites will be capable of complex orbital operations, such as capturing tumbling defunct spacecraft. A fundamental requirement for the success of such missions is the development of accurate spacecraft pose estimation, which provides the servicer's guidance and control system with precise information about the target spacecraft's attitude. This paper addresses the study of such a pipeline using deep learning and classical computer vision algorithms.

Introduction

The accurate estimation of the relative position and attitude (i.e., pose) of a spaceborne object using minimal hardware is an enabling technology for current and future on-orbit servicing and debris removal missions [1]. These missions play a crucial role in mitigating the growing problem of space debris in Low-Earth Orbit (LEO) [2]. Among them are RemoveDEBRIS by the Surrey Space Centre [3], Clearspace-1 by the Swiss start-up 'Clearspace SA' [4], CRD2 by JAXA [5], and EROSS by Thales Alenia Space [6]. Efficient mission planning for such operations requires trajectory optimization techniques, especially when considering refueling or resupply scenarios involving in-situ resource utilization (ISRU) on celestial bodies like the Moon [7]. These complex maneuvers necessitate not only meticulously planned trajectories but also real-time awareness of the target spacecraft's pose throughout the mission. By providing crucial information about the target spacecraft's pose relative to the servicing spacecraft, pose estimation systems serve as the critical "eyes" for the mission. This information is continuously fed into the guidance system. The guidance system utilizes the pose data alongside the pre-programmed mission plan and real-time sensor data to constantly adjust the spacecraft's thrusters, ensuring it stays on the optimized trajectory and executes maneuvers with high precision [8]. Depending on the scenario, the target spacecraft can be either cooperative, utilizing radio links or fiducials, or non-cooperative with known or unknown geometry. Recently, non-cooperative target pose estimation has gained interest from the aerospace society due to the accumulation of inactive satellites and debris in low Earth orbit. However, the current go-to solution for performing pose estimation of target spacecraft involves cumbersome and expensive LIDARs, which will likely hinder the widespread adoption of space debris removal technology. One of the most promising alternatives to LIDARs to date is the use of monocular cameras combined with computer vision algorithms, providing a cheap and lightweight solution. To address this, ESA organized the international Satellite Pose Estimation competition (SPEC2021), the main objective of which was to find the most efficient way to estimate the pose of known uncooperative spacecraft using monocular cameras [9]. SPEC2021 exploits the next-generation spacecraft pose estimation dataset (SPEED+), consisting of labeled images from computer graphics and real-life sources. To solve this problem, each participant

implemented deep learning (DL) techniques as opposed to classical image processing methods. This choice stems from the fact that the latter are too computationally expensive for on-board processors and are not fully robust to the complex space environment, such as harsh lighting conditions. Neural networks, on the other hand, are able to learn complex features that are sometimes impossible to hardcode by humans. However, when trained on synthetic data, DL models typically fail when tested on real images, leading to the so-called domain gap problem [9].

This paper proposes a DL-based pipeline for the pose estimation of uncooperative target spacecraft with known geometry in the pre-capture phase prior to docking. Our method is based on state-of-the-art pose estimation network (YOLOv8-pose) [10] and Perspective-n-Point (PnP) solvers [11]. PnP solvers are well-established algorithms that efficiently calculate the 3D pose of an object in a scene given its corresponding 2D keypoints and a 3D model of the object. The scenario considered in this paper is a synthetic video stream of the client uncooperative spacecraft approaching the servicer until reaching a minimum relative distance of 20 cm, which corresponds to the length of the servicer gripper. To generate a realistic training dataset for the deep learning network we utilized Blender, a popular open-source 3D creation suite [12], and CAD model of the Envisat spacecraft [13]. The Envisat model in Blender was then animated to simulate the approach of a client spacecraft towards a servicer spacecraft, mimicking the pre-capture phase prior to docking.

Methods

Data preparation. The generation of synthetic images is realized through the open-source 3D graphical tool Blender. Firstly, this choice is made because of its integrated Cycles rendering engine, which is physically based and uses ray tracing. Secondly, Blender offers a built-in Python API, enabling control over the software via Python scripting mode. Another useful tool utilized is the third-party Starfish Python library, which facilitates the automatic generation of thousands of images along with pose information for the object in each image. Users can define the object's pose or have it randomly distributed across frames. Moreover, the Starfish library allows for the variation of parameters such as background and lighting orientation in each generated image. After the image dataset is generated with corresponding annotations regarding the position and orientation of the spacecraft, it is necessary to preprocess the dataset and add additional information to the annotation file to enable its use in training neural networks. Specifically, knowing the 3D coordinates of the spacecraft and its pose information, it is possible to retrieve the pixel coordinates of each keypoint using perspective projection equation:

$$\begin{bmatrix} u_i w_i \\ v_i w_i \\ w_i \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} [R(\mathbf{q_{BC}}) \; \mathbf{t_{BC}}] \begin{bmatrix} x_B \\ y_B \\ z_B \end{bmatrix} \quad (1)$$

where:

$- x_B, y_B, z_B$ are the 3 d coordinates of the spacecraft keypoint

$- [R(\mathbf{q_{BC}})\mathbf{t_{BC}}]$ is roto-translation vector

$- f_x, f_y, c_x, c_y$ are the parameters of the intrinsic matrix

$- u_i w_i, v_i w_i$ are the pixel coordinates of the spacecraft key-point

Pose estimation pipeline. The pipeline can be summarized in 2 following steps according to the figure 1:

1. Deployment of Keypoint Regression Network (KRN). In this paper we decided to leverage ultralytics API which uses YOLOv8-pose neural network architecture. It gives the users to choose

between different model size and can be easily transformed to openvvino format for the faster inference on the low-power compute devices. The KRN yields a $1 \times 2N$ vector encoding the 2D positions of N keypoints.

2. Retrieved 2D keypoints together with the available 3d wireframe model of the spacecraft can be fed into off the-shelf PnP solver in order to calculate the position and orientation of the spacecraft.

The selected ground-truth keypoints were strategically chosen to ensure their presence within the field of view (FOV) of the monocular camera. Moreover, a minimum of four points is required to execute the PnP (Perspective-n-Point) algorithm effectively.
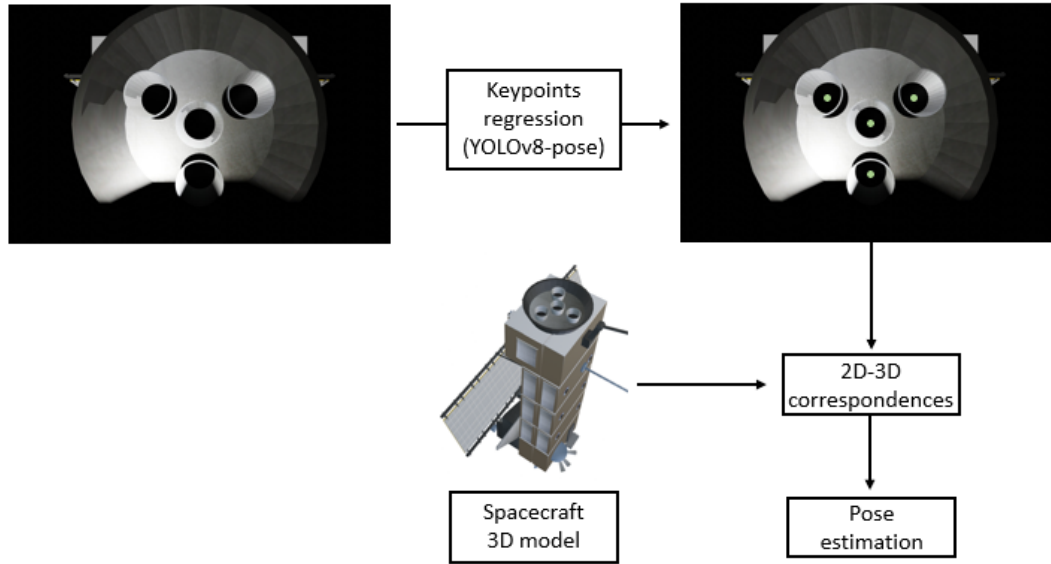

Figure 1: 2-stage pose estimation pipeline.

On the figure 2 there is visualization of some images from the validation test set. It is evident that the KRN based on YOLOv8-pose architecture is able to output the right posses with the very high probability.
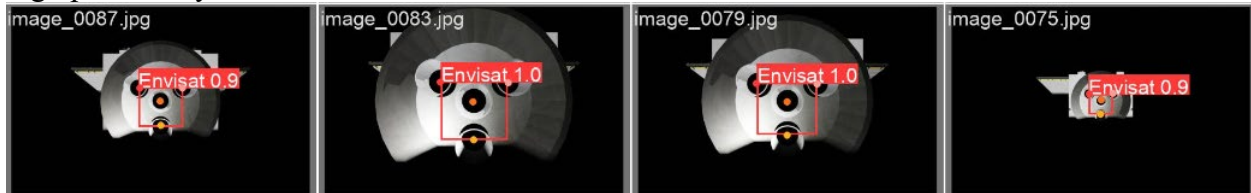

Figure 2: Samples from validation batch during the training of KRN.

The retrieved poses of the spacecraft are compared to the ground truth data according to the metrics from the SPEC2021 competition:

$$E = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{\left|t_{BC_i} - \hat{t}_{BC_i}\right|_2}{\left|t_{BC_i}\right|_2} + 2\arccos(|\langle q_i, \hat{q}_i\rangle|)\right) \qquad (2)$$

Where t_{BC_i}, q_i and $\hat{t}_{BC_i}$, $\hat{q}_i$ are the ground truth and estimated position vectors and attitude quaternions respectively.

Results
A dataset comprising 5000 images from Envisat, each with a resolution of 1920×1200, was created. This dataset was then partitioned into three subsets: a training set containing 3500 images,

a validation set containing 1000 images, and a test set containing 500 images. Considering the limitations of low-power on-board spacecraft processors, we opted for the smallest YOLOv8n-pose network model available from the ultralytics API (See table 1).

Table 1: Characteristics of YOLOv8-pose models with different sizes

Model	size (pixels)	mAP^{pose} 50-95	mAP^{pose} 50	Speed CPU ONNX (ms)	Speed A100 TensorRT (ms)	params (M)	FLOPs (B)
YOLOv8n-pose	640	50.4	80.1	131.8	1.18	3.3	9.2
YOLOv8s-pose	640	60.0	86.2	233.2	1.42	11.6	30.2
YOLOv8m-pose	640	65.0	88.8	456.3	2.00	26.4	81.0

Training was performed on Nvidia GeForce RTX 3090 for 100 epochs with the minimum train and validation pose loses of 0.055 and 0.033 respectively (figure 3). Batch size was set of 32 images.
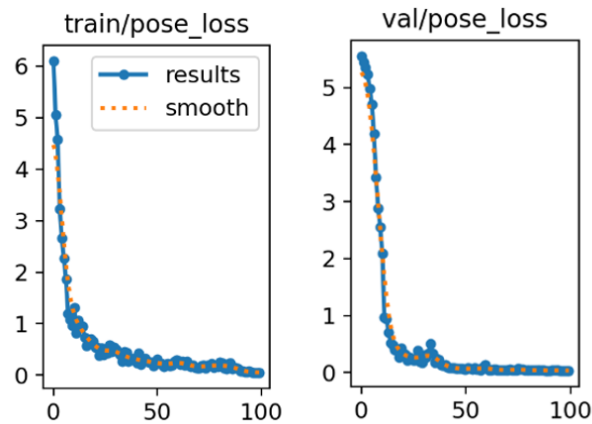

Figure 3: Train and validation pose losses.

The predicted keypoints for each image frame were then fed into the PnP solver to retrieve the poses for the spacecraft across the test set of 500 images dedicated to final validation. Preliminary results are presented in Table 2 and will be improved upon in future work.

Table 2: Evaluation metrics according to the SPEC2021

Mean translation error e_t (m)	0.01907
Mean rotation error e_q (deg)	14.89
Mean SPEED score E	0.29

References

[1] T. H. Park, S. Sharma, and S. D'Amico, "Towards robust learning-based pose estimation of noncooperative spacecraft," arXiv preprint arXiv:1909.00392, 2019

[2] Pietro, D.M. A Ptolemaic Approach Improving the Conjunction Analysis Pipeline for Leo. Aerotec. Missili Spaz. 102, 309–321 (2023). https://doi.org/10.1007/s42496-023-00164-7

[3]   J. L. Forshaw, G. S. Aglietti, N. Navarathinam, H. Kadhem, T. Salmon, A. Pisseloup, E. Joffre, T. Chabot, I. Retat, R. Axthelm, et al., "Removedebris: An in-orbit active debris removal demonstration mission," Acta Astronautica, vol. 127, pp. 448–463, 2016. https://doi.org/10.1016/j.actaastro.2016.06.018

[4]   R. Biesbroek, S. Aziz, A. Wolahan, S.-f. Cipolla, M. Richard-Noca, and L. Piguet, "The clearspace-1 mission: Esa and clearspace team up to remove debris," in Proc. 8th Eur. Conf. Sp. Debris, pp. 1–3, 2021.

[5] T. Yamamoto, J. Matsumoto, H. Okamoto, R. Yoshida, C. Hoshino, and K. Yamanaka, "Pave the way for active debris removal realization: Jaxa commercial removal of debris demonstration (crd2)," in 8th European Conference on Space Debris, p. 200, 2021.

[6]   V. Dubanchet, S. Andiappane, P. Negro, D. Casu, A. Giovannini, G. Durand, and J. D'Amico, "Validation and demonstration of eross project: The european robotic orbital support services," in Proceedings of the International Astronautical Congress, IAC, 2020.

[7] Siena, A. Orbit/Attitude Control for Rendezvous and Docking at the Herschel Space Observatory. Aerotec. Missili Spaz. 103, 39–49 (2024). https://doi.org/10.1007/s42496-023-00188-z

[8] Atmaca, D., Pontani, M. Near-Optimal Feedback Guidance for Low-Thrust Earth Orbit Transfers. Aerotec. Missili Spaz. (2024). https://doi.org/10.1007/s42496-023-00193-2

[9]   T. H. Park, M. Märtens, M. Jawaid, Z. Wang, B. Chen, T.-J. Chin, D. Izzo, and S. D'Amico, "Satellite pose estimation competition 2021: Results and analyses," Acta Astronautica, vol. 204, pp. 640–665, 2023. https://doi.org/10.1016/j.actaastro.2023.01.002

[10] Information on https://docs.ultralytics.com/tasks/pose/

[11] Lu, Xiao. (2018). A Review of Solutions for Perspective-n-Point Problem in Camera Pose Estimation. Journal of Physics: Conference Series. 1087. 052009. https://doi.org/10.1088/1742-6596/1087/5/052009

[12] Information on https://www.blender.org/

[13] CAD model: https://sketchfab.com/3d-models/envisat-65b0ec49681a44f68dfc8bd4efe9583