SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Executive Summary of the Thesis

Deep Learning Solutions for Attitude Ambiguity in Relative Navigation with Unknown and Uncooperative Targets

Laurea Magistrale in Space Engineering - Ingegneria Spaziale

Author: Matteo Rosa

Advisor: Michele Maestrini

Co-advisor: Pierluigi Di Lizia, Maria Alessandra De Luca, Niccolò Faraco

Academic year: 2022-2023

1. Introduction

With the expansion space activities, the number of space debris in LEO orbits has been growing in recent years, thus increasing the risk of possible collisions with spacecrafts. To tackle this issue, two activities, Active Debris Removal (ADR) and In-Orbit Servicing (IOS), have gained importance. The execution of these activities requires operating in proximity of Resident Space Object (RSO), making relative navigation a critical aspect. This problem has been extensively studied in the literature, and has been differentiated based on the nature of the target. Targets are classified considering their level of cooperation and available information. Cooperative targets have the ability to communicate, actively or passively, with the chaser, while non-cooperative targets lack this capability. If a priori information about the targets is available on board, they are classified as known targets, otherwise they are classified as unknown. The most difficult category of targets to deal with is the one of unknown and non-cooperative targets. Current approaches employ model-based methods to address this specific category. One such method, CoMBiNa, introduced in [5], offers a reliable estimation for asymmetric targets but encounters limitations when dealing with symmetric targets. An expansion of this approach is introduced in [1], to tackle the challenge posed by symmetrical targets. This extension involves approximating the target as a gyroscopic structure, enabling modification of the state vector concerning the attitude, thus allowing at least the orientation of the axis of symmetry of the target to be correctly estimated.

The aim of the first part of this work is to modify the CoMBiNa algorithm to estimate the complete attitude of the symmetric target, rather than just the orientation of the axis of symmetry. This involves integrating CoMBiNa with an operational neural network, capable of supplying the precise locations of specific target features. Subsequently, in the second part of the work, the feasibility of using such a neural network to identify the locations of specific features on the target is demonstrated.

2. Introduction to CoMBiNa

CoMBiNa comprises two distinct phases called the preliminary and operational phases. In the preliminary phase, the chaser operates within a safe orbit with the aim of reconstructing the tar-

get model using SLAM technique. The acquisition of images for this reconstruction is carried out using a stereo camera. During the operational phase, the objective is to estimate the relative position $\boldsymbol{r}$ and velocity $\boldsymbol{v}$, and the entire attitude of the target ($\boldsymbol{\sigma}$,$\boldsymbol{\omega}$,$\boldsymbol{k}$). To do this, the acquired images are processed and matched with the coarse model, obtained in the previous phase, using the Bayesian Coherent Point Drift (BCPD) algorithm, which is a statistical point-set registration technique. The information derived from BCPD provides the measurements for two Unscented Kalman Filters (UKFs). The first UKF focuses on reconstructing the relative position and velocity between the chaser and the target (translational filter), while the second UKF concentrates on reconstructing the attitude of the target (rotational filter). As indicated by the results presented in [5], the algorithm encounters failure when dealing with a symmetrical target. The issue arises due to the unobservable rotations around the axis of symmetry, leading BCPD to converge to incorrect relative pose solutions.
To address this limitation, in [1], the target is approximated as a gyroscopic structure. By reformulating the attitude state vector, it becomes feasible to derive the orientation of the symmetry axis of the target. The reformulation requires the state vector to contain the coordinates of the symmetry axis of target $\boldsymbol{J}$ in the inertial frame $\mathcal{I}_\Gamma$ and $\frac{\Gamma_{(0)}}{A}$. $\boldsymbol{J}$ can be expressed by just two angles in spherical coordinates (azimuth, elevation). This reduces the number of variables from four to three. The new state vector is outlined as follows:

$$\hat{\mathbf{x}}_a = \begin{bmatrix} \phi \\ \delta \\ \frac{\Gamma_{(0)}}{A} \end{bmatrix} \tag{1}$$

The new pipeline of the operational phase of CoMBiNa is shown in the Fig.1. As far as angular velocity is concerned, the only information that can be obtained is the absolute value of the equatorial component of the angular velocity and the square of the ratio between the angular velocity about the symmetry axis and the moment of inertia A about the other two axes.
Employing this revised parameterization requires the introduction of a rigid rotation matrix between the inertial system $\mathcal{I}_\Gamma$ and the general inertial system $\mathcal{I}$. Initially, in [1], this matrix was assumed to be known; subsequently, the analysis was reiterated with consideration for its unknown status. In the latter scenario, the MRPs ($\boldsymbol{\sigma}_{\mathcal{I}_\Gamma \mathcal{I}}$) of this matrix were incorporated into the attitude state vector.

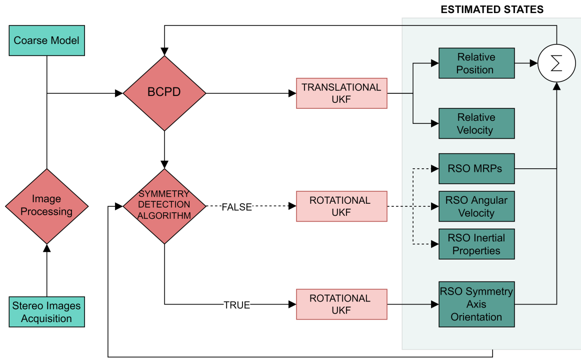

Figure 1: Extended CoMBiNa operational phase (courtesy of [1])

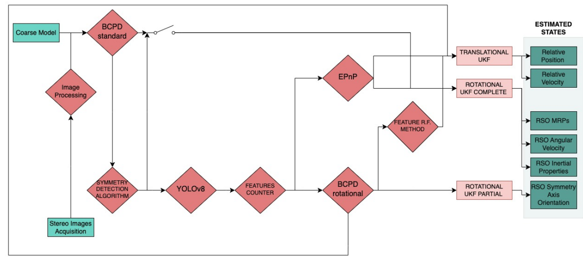

Figure 2: Pipeline for the CoMBiNa operational phase

3. CoMBiNa: New formulation

In the new CoMBiNa formulation, the aim is to estimate the full pose (MRPs $\boldsymbol{\sigma}_{\mathcal{T}}$, angular velocity $\boldsymbol{\omega}_{\mathcal{T}}$ and the inertial parameters $\boldsymbol{k}$) of the symmetrical target. This new process assumes the presence of features on the outer surface of the target, such as stickers representing various state flags, symbols from space agencies, or luminous LEDs. Furthermore, it is assumed that a neural network is available to detect the positions of the centers of these objects in the images provided by the stereo camera. In the preliminary phase, where the satellite model reconstruction occurs, the positions of features on the model are concurrently reconstructed, ensuring their availability onboard. The new pipeline of the operational phase of CoMBiNa is shown in Fig. 2.

The changes made to the previous formulation of CoMBiNa are both in the *Main Loop* algorithm, which includes the procedure for updating and propagating the states and covariance matrices using UKFs, and in the *Optimization Function* algorithm, which describes the procedure used to derive the measurements used by the filters. In the *Main Loop*, three states are used instead of two:

$$\hat{\mathbf{x}}_s = [\hat{\boldsymbol{r}}; \hat{\boldsymbol{v}}], \hat{\mathbf{x}}_{a,1} = \left[\hat{\phi}; \hat{\delta}; \frac{\hat{\Gamma}_{(0)}}{A}\right], \hat{\mathbf{x}}_{a,2} = [\hat{\boldsymbol{\sigma}}; \hat{\boldsymbol{\omega}}; \hat{k}] \tag{2}$$

$\hat{\mathbf{x}}_{a,1}$ contains the variables describing the orientation of the symmetry axis of the target. This state vector is only used if there are no features in the image. In this case $\hat{\mathbf{x}}_{a,1}$ is defined, for the first time, considering the last available information about $\hat{\mathbf{x}}_{a,2}$, while $\hat{\mathbf{x}}_{a,2}$ is only propagated by the filter, but its value is not corrected. The most relevant changes lie in the way measurements are acquired, and consequently in the *Optimization Function*. Since in [1] the relative attitude measurement provided by BCPD was unreliable, the aim now is to find an alternative way to obtain it by exploiting the presence of features in the image. The method that is used changes depending on the number of features available in the image.

When a minimum of four features are present in the image, the EPnP method is employed to recover the $\mathbf{C}_{\mathcal{TI}}$ matrix, consequently obtaining the Motion Representation Parameters (MRPs). The EPnP [4] algorithm provides both the rotation $\mathbf{R}_{\mathcal{ST}}$ and translation $\mathbf{T}_{\mathcal{ST}}$ of the camera reference frame with respect to the target reference frame. Using this information, the $\mathbf{C}_{\mathcal{TI}}$ matrix and relative position $\boldsymbol{r}$ can be derived through the following process:

$$\mathbf{C}_{\mathcal{TI}} = \mathbf{R}_{\mathcal{ST}}^T \mathbf{C}_{\mathcal{CS}}^T \mathbf{C}_{\mathcal{CI}} \tag{3}$$

$$\boldsymbol{r} = \mathbf{C}_{\mathcal{CL}}^T \mathbf{C}_{\mathcal{CS}} \mathbf{T}_{\mathcal{ST}} \tag{4}$$

where $\mathbf{C}_{\mathcal{CS}}$ is the rotation of the camera frame with respect to the chaser frame, $\mathbf{C}_{\mathcal{CL}}$ is the rotation of the chaser frame with respect to the LVLH frame and $\mathbf{C}_{\mathcal{CI}}$ is the rotation of the chaser frame with respect to the inertial frame. However, when the number of available points is equal to four, EPnP often yields incorrect results. For this reason, after acquiring measurements from EPnP, the validity of the result is verified by examining the squared Mahalanobis Distance.

In the cases where features are present in the image, but not in sufficient number to use EPnP, or when the measurements provided by EPnP are rejected, the Feature Reference Frame method is used. Initially, BCPD is used to obtain measurements of the relative position and orientation of $\boldsymbol{J}$. Subsequently, a feature is selected, from those whose positions are provided in the image by the neural network, and associated with its corresponding feature in the coarse model. The features of the coarse model were reconstructed during the preliminary phase, along with the model itself, and they are consequently available on board. Using the positions of the two features expressed in the image and in the target reference frame, the feature system is constructed with respect to the inertial system and with respect to the target system. This allows to devise a fast algorithm to retrieve an estimate of $\mathbf{C}_{\mathcal{TI}}$ and the corresponding MRPs.

In the scenario, where no features are available, the algorithm back falls into the situation previously outlined in [1]. The only measurements that can be obtained are the relative position and the orientation of the symmetry axis of the target.

3.1. Results

A numerical analysis was performed to test the new formulation of CoMBiNa. The chaser is considered to be placed on a circular orbit in Low Earth Orbit (LEO) with a radius of 800 km.

This formulation was tested for both known and unknown rigid rotation matrix. In both cases, analyses were performed by varying both the number of features present on the body (1 to 14) and the probability of feature failure detection (0%, 20%, 40%). Convergence of all variables is achieved in both cases. The Fig. 3,4,5,6,7 show the convergence of variables in the worst case analysed, where the rigid rotation matrix is unknown, only one feature is present and the probability of failure is 40%.

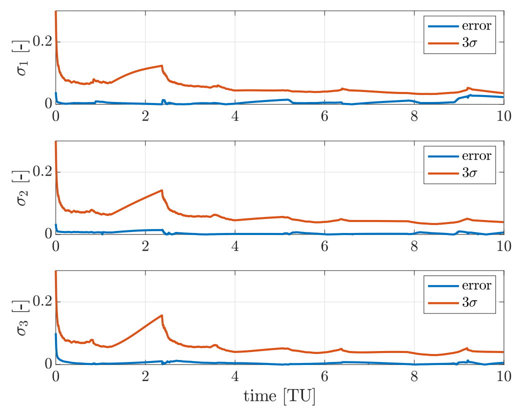

Figure 3: MRPs of the target

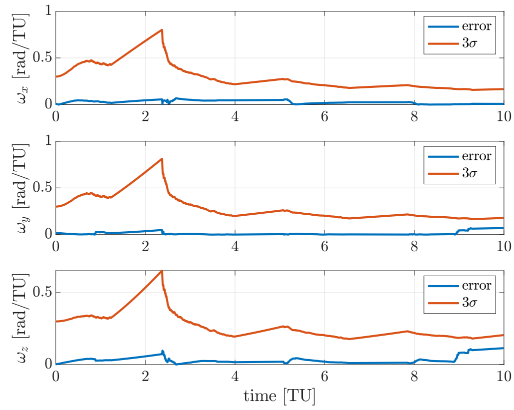

Figure 4: Angular velocity of the target

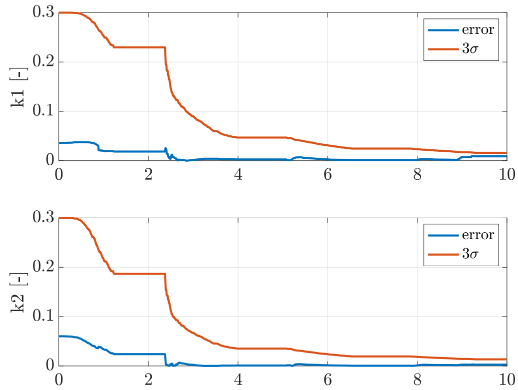

Figure 5: Inertia parameters of the target

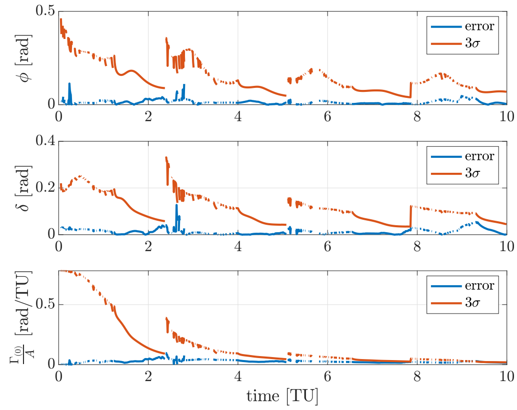

Figure 6: Target's symmetry axis spherical coordinates in $\mathcal{I}_\Gamma$ frame and $\dfrac{\Gamma_{(0)}}{A}$ ratio

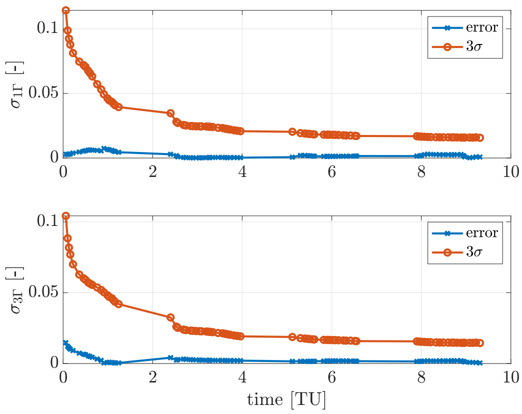

Figure 7: MRPs of the rigid rotation matrix

4. Computer Vision

4.1. Object detection

In order to identify the features on the target spacecraft, a convolutional neural network for object detection is selected. The goal of the object detection is to detect all instances of the predefined classes and provide its coarse localization in the image by axis-aligned boxes. The detector should be able to identify all instances of the object classes, i.e. categories of objects we are interested in, and draw bounding box around it. Various methods exist for object detection, which can be categorized into three main groups: region proposal (two-stage), one-stage and transformer-based detectors. All these techniques rely on Convolutional Neural Networks (CNNs). One-stage detection methods are preferred over region proposal methods due to their superior computational efficiency, a crucial factor enabling real-time operations in this specific application domain. Within the one-stage category, the chosen architecture is You Only Look Once (YOLO), which processes the entire image using a convolutional network. In this study, YOLOv8, introduced by Ultralitics [3] is used. Ultralitics offers five distinct versions, which differ in size, that are: YOLOv8n (nano), YOLOv8s (small), YOLOv8m (medium), YOLOv8l (large), and YOLOv8x (extra-large).

5. Features Detection

The objective now is to illustrate that through neural network training, it is possible to identify specific features, such as a NASA or ESA sticker, on a symmetrical target. These particular features were selected due to the fact that they are very common on satellites, making them ideal candidates to replicate real-world conditions. A CAD model of a satellite and Blender, which is a 3D computer graphics software toolset, are used to generate a sufficiently large dataset of images. After uploading the model to Blender, features were manually placed on the outer surface. It was decided to put flags of some states (Italy, USA, Germany), symbols of some space agencies (ESA and NASA) and luminous LEDs (red and green). The images and text files containing the bounding box information were generated using JINS [2], a tool is able to automate the rendering of images.
Following dataset creation, YOLOv8 underwent training using these images. YOLOv8s was chosen as the pretrained model, as it represents a good trade-off between mAP and execution speed. The training process was carried out on 150 epochs using a patience of 20. Fig. 8 and 9 show the training and validation box and class losses, respectively. From these figures, it can be inferred that the behaviour of the losses is correct and that the model is converging.
By validating the trained model, the mAP_{50} and mAP_{50-95} are calculated, which are:

$$mAP_{50} = 0.953 \qquad mAP_{50-95} = 0.719 \tag{5}$$

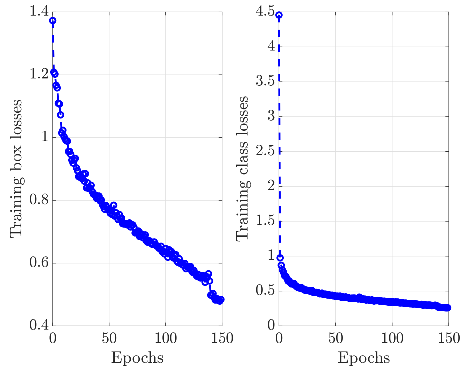

Figure 8: Training box e class losses

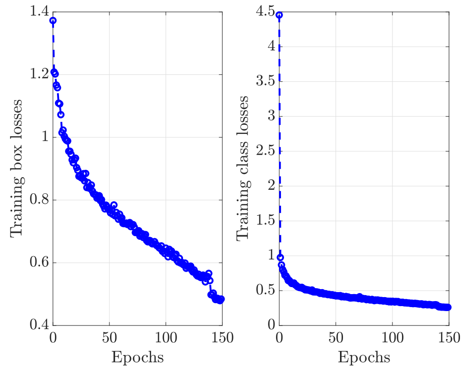

Figure 9: Validation box and class losses

Subsequently, the trained network underwent testing using new images. The network's output, displayed in Fig. 10, illustrates both the bounding box outlining the object and the corresponding confidence level denoting the network's certainty regarding the object's classification within that specific class.

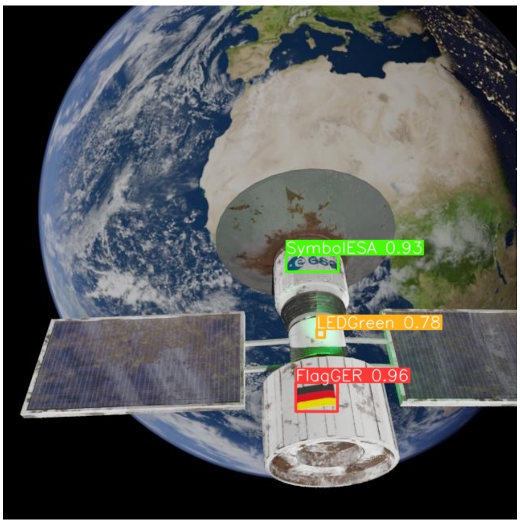

Figure 10: Output image generated by YOLOv8

For each predicted bounding box, errors along both the x-axis of the image (err_x) and the y-axis (err_y) are computed. Subsequently, the mean error and standard deviation concerning both the x and y axes of the image are determined, leading to the following results:

	Mean Value (pixels)	Standard Deviation (pixels)
err_x	0.7342	1.0906
err_y	0.4758	0.9052

Table 1: Mean errors and standard deviations

It was also verified that the failure detection values used in Section 4 were consistent with the results obtained by calculating the failure detection rate for each object class.

6. Conclusions

With the original CoMBiNa [5], the algorithm is not expected to converge when dealing with a symmetric RSO, due to the unobservable rotations about the symmetry axes. A reformulation of CoMBiNa is presented in [1], in which the target is approximated to a gyroscopic object to obtain an estimate of the orientation of its axis of symmetry.
This study aims to achieve a complete determination of the attitude, even in the case of symmetrical targets. To accomplish this, an integration of a neural network into CoMBiNa is implemented, enabling the detection of features positioned on the surface of the target in images obtained from a stereo camera. This process effectively breaks the symmetry of the target, making it possible to estimate its complete attitude.

6.1. Future Works

Considering the navigation part, the next step is to integrate the neural network within CoMBiNa, to avoid simulating feature measurements in the image, and then use those provided by YOLOv8. This integration is useful as certain failure conditions, such as false positives or wrong feature classification, are difficult to be simulated. By integrating the network within the loop, it becomes feasible to introduce these effects during algorithm testing and assess their impact.
This work does not account for other failures which are characteristic of neural networks such as false positives (i.e., identification of features in wrong locations), detection of multiple features of the same type (i.e., introducing an ambiguity), and the possibility of wrong feature classification (i.e., wrong assignment of ID to an identified feature). The introduction of procedures to deal with these problems constitute an interesting future research direction.
Concerning the training and testing of YOLOv8, it proves beneficial to diversify the 3D models employed rather than relying only on a single target, thereby enhancing the diversity of the dataset. Since unknown RSOs are considered in this work, it would be advantageous to evaluate the performance of the network on images containing a target different from the one used during training. Testing the network with a new model could produce slightly worse results.

References

[1] Maria Alessandra De Luca. Relative navigation at unknown uncooperative symmetric objects. 2022.

[2] Niccolò Faraco, Michele Maestrini, and Pierluigi Di Lizia. Instance segmentation for feature recognition on noncooperative resident space objects. *Journal of Spacecraft and Rockets*, 59(6):2160–2174, 2022.

[3] Glenn Jocher, A Chaurasia, and J Qiu. Yolo by ultralytics.

[4] Vincent Lepetit, Francesc Moreno-Noguer, and Pascal Fua. Ep n p: An accurate o (n) solution to the p n p problem. *International journal of computer vision*, 81:155–166, 2009.

[5] Michele Maestrini. Satellite inspection of unknown resident space objects. 2022.