ON-GROUND VALIDATION OF A CNN-BASED MONOCULAR POSE ESTIMATION SYSTEM FOR UNCOOPERATIVE SPACECRAFT

L. Pasqualetto Cassinis[1], A. Menicucci[1], E. Gill[1], I. Ahrns[2], and J. Gil-Fernández[3]

[1]*Delft University of Technology, Kluyverweg 1 2629 HS, Delft, The Netherlands, Email: {L.PasqualettoCassinis, A.Menicucci, E.K.A.Gill}@tudelft.nl*
[2]*Airbus DS GmbH, Airbusallee 1, 28199, Bremen, Germany, Email: ingo.ahrns@airbus.com*
[3]*ESTEC, Keplerlaan 1, 2201 AZ, Noordwijk, The Netherlands, Email: J.Gil.Fernandez@esa.int*

ABSTRACT

The estimation of the relative pose of an inactive spacecraft by an active servicer spacecraft is a critical task for close-proximity operations, such as In-Orbit Servicing and Active Debris Removal. Among all the challenges, the lack of available space images of the inactive satellite makes the on-ground validation of current monocular camera-based navigation systems a challenging task, mostly due to the fact that standard Image Processing (IP) algorithms, which are usually tested on synthetic images, tend to fail when implemented in orbit. This paper reports on the testing of a novel Convolutional Neural Network (CNN)-based pose estimation pipeline with realistic lab-generated 2D monocular images of the European Space Agency's Envisat spacecraft. Following the current need to bridge the reality gap between synthetic and images acquired in space, the main contribution of this work is to test the performance of CNNs trained on synthetic datasets with more realistic images of the target spacecraft. The validation of the proposed pose estimation system is assured by the introduction of a calibration framework, which ensures an accurate reference relative pose between the target spacecraft and the camera for each lab-generated image, allowing a comparative assessment at both keypoints detection and pose estimation level. By creating a laboratory database of the Envisat spacecraft under space-like conditions, this work further aims at facilitating the establishment of a standardized on-ground validation procedure that can be used in different lab setups and with different target satellites. The lab-representative images of the Envisat are generated at the Orbital Robotics and GNC lab of ESA's European Space Research and Technology Centre (ESTEC). The VICON Tracker System is used together with a KUKA robotic arm to respectively track and control the trajectory of the monocular camera around a scaled 1:25 mockup of the Envisat spacecraft.

Keywords: Convolutional Neural Networks; On-ground Validation; Monocular Pose Estimation; Calibration Procedure.

1. INTRODUCTION

Nowadays, the safety and operations of satellites in orbit has become paramount for key Earth-based applications, such as remote sensing, navigation, and telecommunication. In this context, advancements in the field of Guidance, Navigation, and Control (GNC) were made in the past years to cope with the challenges involved in In-Orbit Servicing (IOS) and Active Debris Removal (ADR) missions [27, 29]. For such scenarios, the estimation of the relative pose (position and attitude) of an uncooperative spacecraft by an active servicer spacecraft represents a critical task. Compared to cooperative close-proximity missions, the pose estimation problem is indeed complicated by the fact that the target satellite is not functional and/or not able to aid the relative navigation. Hence, optical sensors shall be preferred over Radio Frequency (RF) sensors to cope with a lack of navigation devices such as Global Positioning System (GPS) sensors and/or antennas onboard the target.

In this framework, pose estimation systems based solely on a monocular camera are recently becoming an attractive alternative to systems based on active sensors or stereo cameras, due to their reduced mass, power consumption and system complexity [23, 14]. However, a significant effort is still required to comply with most of the demanding requirements for a robust and accurate monocular-based navigation system. Notably, the aforementioned navigation system cannot rely on known visual markers, as they are typically not installed on an uncooperative target. Since the extraction of visual features is an essential step in the pose estimation process, advanced Image Processing (IP) techniques are required to extract keypoints (or interest points), corners, and/or edges on the target body. In model-based methods, the detected features are then matched with pre-defined features on an offline wireframe 3D model of the target to solve for the relative pose. This is usually achieved by solving the Perspective-n-Points (PnP) problem [21]. In other words, a reliable detection of key features under adverse orbital conditions is highly desirable to guarantee safe operations around an uncooperative spacecraft. Unfortunately, standard IP algorithms usually lack of fea-

ture detection robustness when applied to space images [3], undermining the overall navigation system and, in turn, the whole close-proximity operations around the uncooperative target. From a pose initialization standpoint, the extraction of target features can in fact be jeopardized by external factors, such as adverse illumination conditions, low Signal-to-Noise ratio (SNR) and Earth in the background, as well as by target-specific factors, such as the presence of complex textures and features on the target body. Moreover, most of the IP methods are based on the image gradient, detecting textured-rich features or highly visible parts of the target silhouette. As such, the detected features are image-specific and can vary in number and typology depending on the image histogram. This means that most of these techniques cannot accommodate an offline feature selection step, which translates into a computationally expensive image-to-model correspondence process to ensure that each detected 2D feature is matched with its 3D counterpart on the available wireframe model of the target object.

In recent years, Convolutional Neural Networks (CNNs) are emerging as a valid and robust alternative to standard monocular-based pose estimation systems, with two main CNN-based architectures currently being investigated. Initially, *end-to-end* architectures in which a single CNN replaced the entire pose estimation pipeline were more exploited [20, 22, 24, 25]. However, since the pose accuracies of these systems proved to be lower than the accuracies returned by standard PnP solvers, especially in the estimation of the relative attitude [20], keypoints-based architectures stood out as the preferred option. Specifically, average orientation errors of $1.31^o \pm 2.24^o$ were achieved by keypoints-based methods as opposed to the average orientation errors of $9.76^o \pm 18.51^o$ achieved by end-to-end methods. These averages were computed across test images of the TANGO spacecraft as part of the Spacecraft Pose Estimation Dataset challenge [8]. In keypoints-based CNN systems, a CNN is used only at a feature detection level to replace standard IP algorithms, and the output features are fed to a PnP solver together with their body coordinates, which are made available through the wireframe 3D model of the target body. Due to the fact that the trainable features can be selected offline prior to the training, the matching of the extracted feature points with the features of the wireframe model can be performed without the need of a large search space for the image-model correspondences, which usually characterizes most of the edges/corners-based methods [3]. However, due to a lack of availability of representative space images, these CNN systems often need to be trained with synthetic renderings of the available target model. As a result, their feature detection robustness on more realistic images is usually unknown and difficult to predict.

In this context, the on-ground validation of the CNNs' performance shall be sought by testing their robustness against representative images of the target spacecraft, generated in a laboratory environment which recreates space-like illumination conditions. In this way, the performance of synthetically-trained CNNs on lab-generated images can be tested. Moreover, a calibration framework shall be established which returns an accurate reference for the relative pose between the monocular camera and the target mockup for each generated image, in order to be able to quantify the CNN performance at both keypoints detection and pose estimation levels.

Several laboratory setup exist to recreate rendezvous approaches around a mockup of a target spacecraft with a monocular camera [30], e.g. the Space Rendezvous Laboratory (SLAB) at Stanford University [8], the Orbital Robotics & GNC laboratory (ORGL) at the European Space Research and Technology Centre (ESTEC) [31], and the Testbed for Robotic Optical Navigation (TRON) at the German Aerospace Agency (DLR) [9]. However, only a few detailed calibration procedures were recently described which allow the accurate estimation of the reference relative pose between camera and target [28]. Besides, the calibration of the target spacecraft highly depends on the presence (cooperative target) or not (uncooperative target) of visual markers, as well as on the rendezvous trajectory that shall be recreated (static or rotational target). As a result, the existing calibration procedures usually require adaptations to the specific setup.

In relation to the on-ground validation of CNN-based pose estimation systems, an additional challenge also arises from bridging the gap between the synthetic renderings and the lab-representative images. If the synthetic dataset used to train the CNN fails in representing the textures of the target mockup as well as the specific illumination in the laboratory setup, the performance on lab-generated images will in fact result in inaccurate detections and lead to low pose estimation accuracies. To overcome this, recent works addressed the impact of augmented synthetic datasets on the CNN performance in either lab-generated or space-based imagery [13, 1]. These augmented datasets are built on a backbone of purely synthetic images of the target by adding noise, randomized and real Earth background, and randomized textures of the target model. However, the synthetic and laboratory environment are usually tuned to return a high representativeness of the synthetic images. Furthermore, the same 3D model is usually used in bot the synthetic rendering and in the laboratory setup. As a result, the CNN detection robustness against variations in the target model has not been fully addressed yet.

In this framework, the main objectives of this paper are:

- To propose a calibration procedure capable of estimating accurate reference poses between the monocular camera and the target spacecraft
- To investigate the impact of datasets augmentation and randomization on the CNN training, validation and testing
- To improve the performance of synthetically-trained CNNs on lab-generated images.

Specifically, the main novelty of this work is to inves-

tigate the performance of the proposed pose estimation system when the mockup of the target spacecraft differs from the rendering model used to synthetically-train the CNN.

The paper is organized as follows. Section 2 introduces the proposed pose estimation framework. The laboratory setup and the calibration procedure are described in Sections 3-4. In Section 5, the CNN training, validation and testing phases are detailed. Special focus is given to the augmentation and randomization pipeline. Section 6 illustrates the adopted pose estimation methods, whereas the results are presented in Section 7. Finally, Section 8 provides the main conclusions and recommendations.

2. POSE ESTIMATION FRAMEWORK

From a high-level perspective, a model-based monocular pose estimation system receives as input a 2D image and matches it with an existing wireframe 3D model of the target spacecraft to estimate the pose of such target with respect to the servicer camera. Referring to Figure 11, the pose estimation problem consists in determining the position of the target's centre of mass $\mathbf{t}^C$ and its orientation with respect to the camera frame C, represented by the rotation matrix $\mathbf{R}_B^C$. The Perspective-n-Points (PnP) equations,

$$\mathbf{r}^C = \begin{pmatrix} x^C & y^C & z^C \end{pmatrix}^T = \mathbf{R}_B^C \mathbf{r}^B + \mathbf{t}^C \qquad (1)$$

$$\mathbf{p} = (u_i, v_i) = \left(\frac{x^C}{z^C} f_x + c_x, \frac{y^C}{z^C} f_y + c_y \right), \qquad (2)$$

relate the unknown pose with a feature point $\mathbf{p}$ in the image plane via the relative position $\mathbf{r}^C$ of the feature with respect to the camera frame. Here, $\mathbf{r}^B$ is the point location in the 3D model, expressed in the body-frame coordinate system B, whereas f_x and f_y denote the focal lengths of the camera and (C_x, C_y) is the principal point of the image.

From these equations, it can be seen that an important aspect of estimating the pose resides in the capability of the IP system to extract features $\boldsymbol{p}$ from a 2D image of the target spacecraft, which in turn need to be matched with pre-selected features $\boldsymbol{r}^B$ in the wireframe 3D model. Notably, such wireframe model of the target needs to be made available prior to the estimation.

The on-ground validation pipeline of the proposed pose estimation system is shown in Figure 2 and consists of the following main stages:

1. **Calibration procedure and Image Acquisition**: laboratory images of a scaled 1:25 mockup model of the Envisat spacecraft are generated by mounting the camera on a robotic arm which performs a trajectory around the mockup. Besides, the camera is intrinsically and extrinsically calibrated with respect to the Envisat mockup in order to associate reference labels of the relative pose between the adopted monocular camera and the mockup for each generated image

2. **Dataset Generation and CNN Training**: a keypoints-based CNN is trained and validated on augmented datasets. The augmentation is performed by introducing image noise, artificial lights, random background and random textures into synthetically-generated images of a rendering model of the Envisat spacecraft

3. **Online Inference**: the keypoints-based CNN is tested on both synthetic and lab-generated images. The relative pose is estimated by feeding a PnP solver with the detected keypoints as well as with the intrinsic camera parameters and 3D model of Envisat

4. **Validation of Pose Estimation Results**: the CNN-based pose estimation results on the lab-generated images are validated against the reference pose labels, derived from the calibrated objects.

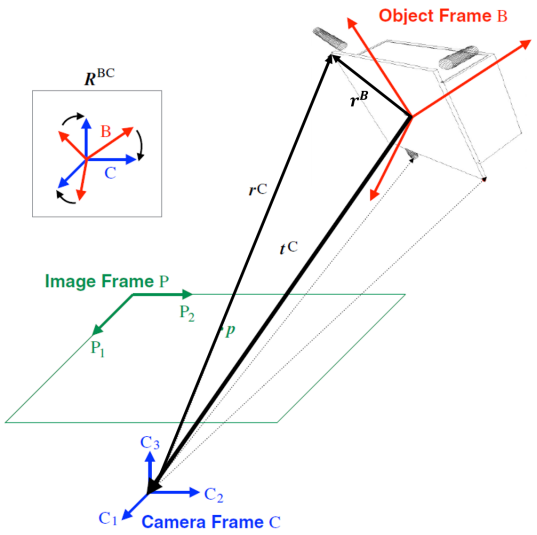


Figure 1. Schematic of the PnP problem using a monocular image (Figure adapted from [23]).

3. THE ORGL FACILITY

The adopted laboratory setup is illustrated in Figure 3 and makes use of the GNC Rendezvous, Approach and Landing Simulator (GRALS) testbed of the ORGL facility at ESTEC. The setup is constituted of the following elements: (a) a 1:25 scaled mockup of the Envisat spacecraft mounted on a black-painted, static tripod; (b) a Prosilica GT4096 monocular camera mounted on a fixed aluminum plate; (c) a ceiling KUKA robotic arm, used to move the camera around the mockup; (d) the VICON

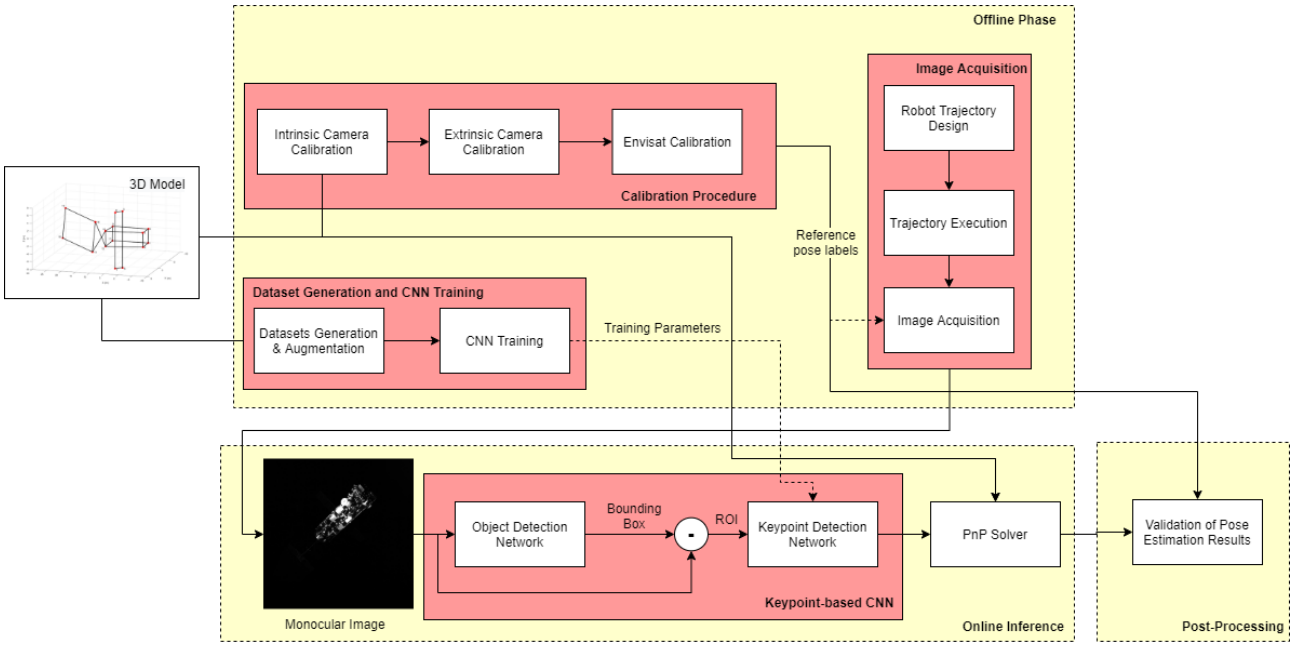


Figure 2. Illustration of the proposed on-ground validation of the CNN-based pose estimation system.

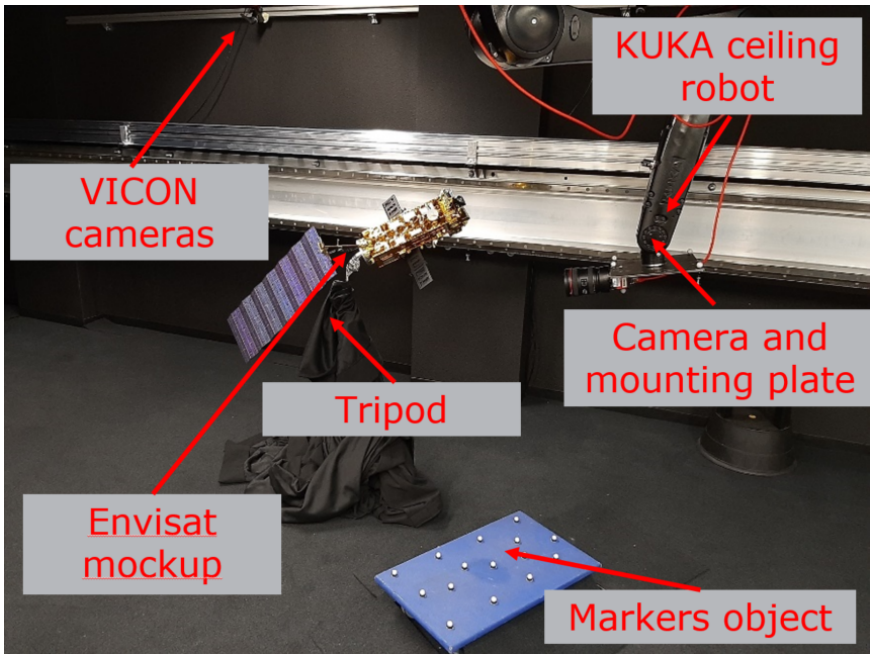


Figure 3. GRALS facility with the scaled 1:25 Envisat mockup, the VST and the monocular camera mounted on the KUKA robotic arm. One of the VTS cameras and the markers object used in the extrinsic calibration of the camera are also shown.

Tracker System (VTS), used to track objects with retro-reflective markers and to provide estimates of their pose with respect to a user-defined reference frame; (e) an external computer providing the software interface between the monocular camera, the VTS and the KUKA robotic arm.

3.1. VICON Tracking System

The VTS is a highly accurate motion capture system capable of tracking dynamic objects with millimeter accuracy [11]. The system includes a set of calibrated IR cameras, some retro-reflecting spherical markers which can be detected and tracked by the cameras, and a software interface to stream telemetry to the external computer. In the current setup, a subset of 10 cameras is selected such that the total field of view covers the operating volume in which the image acquisition is carried out.

3.2. KUKA Software and Hardware Elements

The KUKA robotic arm is controlled from the external computer via a Robot Software Interface (RSI) connection. The arm can translate along a ceiling rail and rotate around its six joints, thus guaranteeing the execution of an elliptical trajectory around the Envisat mockup at around 1-2 m distance.

4. CALIBRATION FRAMEWORK

The calibration setup consists of the elements described in Section 3 and is inspired by the calibration procedure reported in [28]. The objective is to estimate the relative pose between the monocular camera and the Envisat mockup for each generated image.

4.1. Reference Frames Definition

Referring to Figure 4, the following reference frames are defined:

- *VRT Reference Frame O*: this is the reference frame in which all the objects tracked by VTS are expressed
- *Camera Frame C*: this frame is defined such that the third axis is perpendicular to the image plane and is aligned with the optical axis of the camera, with the other two axes planar to the focal plane of the camera
- *Plate Reference Frame I*: this reference frame is built from retro-reflective VTS markers and is rigidly attached to the camera mounting plate

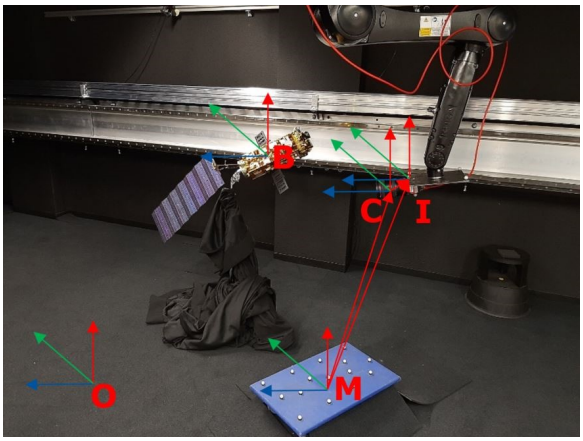


Figure 4. Illustration of the reference frames adopted during the calibration procedure.

- *Envisat Body Frame B*: this is a rigid frame oriented with its axis parallel to the principal axes of inertia of the Envisat mockup and centered on the service module
- *Markers Object Frame M*: this frame is built from retro-reflective VTS markers attached to a planar surface

The transformation between each of these frames can be expressed by a roto-translation matrix $\boldsymbol{T}$, which incorporates the relative rotation matrix $\boldsymbol{R}$ and the relative position vector $\boldsymbol{t}$,

$$\boldsymbol{T} = \begin{pmatrix} \boldsymbol{R} & \boldsymbol{t} \\ \boldsymbol{0}_{1\times 3} & 1 \end{pmatrix}. \tag{3}$$

4.2. Camera Intrinsic Calibration

The first step of the calibration procedure consists of the estimation of the camera intrinsic parameters, such as the focal length, the principal point and the tangential and radial distortion coefficients. This is accomplished by taking images of a chessboard with different camera views and using the *estimateCameraParameters* Matlab built-in function. The function estimates for the intrinsic parameters and the distortion coefficients of a single camera, whilst also returning the images used to estimate the camera parameters and the standard estimation errors for the single camera calibration.

4.3. Extrinsic Calibration

Once the camera intrinsic parameters are estimated, the relative roto-translation matrix $\boldsymbol{T}_C^B$ between the camera frame C and the Envisat body frame B shall be estimated. The procedure consists of the following steps:

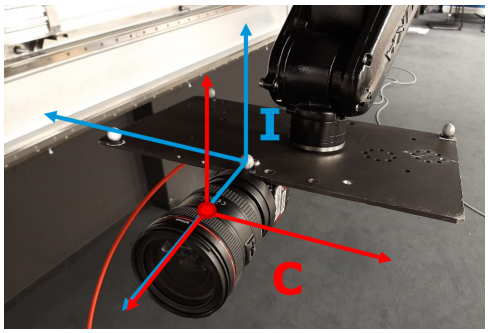


Figure 5. Illustration of the location of reference frame I with respect to the camera frame C. Note that the exact location of the C frame is unknown prior to calibration.

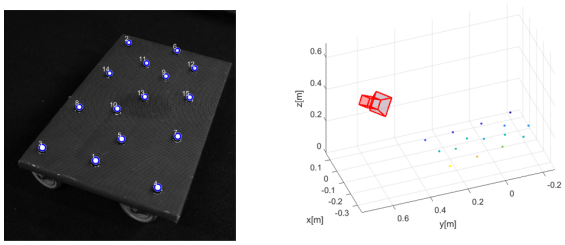
- Estimation of the roto-translation matrix $\boldsymbol{T}_C^I$ - Camera Extrinsic Calibration
- Estimation of the roto-translation matrix $\boldsymbol{T}_O^B$ - Mockup-to-VTS Calibration
- Estimation of the roto-translation matrix $\boldsymbol{T}_B^C$ - Mockup-to-Camera Calibration

4.3.1. Estimation of the roto-translation matrix $\boldsymbol{T}_C^I$

The first task is to recreate the objects M and I in Figure 4 by placing some retro-reflective markers onto the camera mounting plate and a planar surface, respectively. Based on similar setups [28], 15 markers were used to recreate the object M, whereas a total of 9 markers were chosen for the camera mounting plate in order to guarantee a reliable tracking by the VTS throughout the whole image acquisition trajectory. Figure 5 illustrates the location of the I frame with respect to the camera frame C. Only four out of the nine markers are shown for clarity.

Next, the planar object M is moved in order to generate pictures of the retro-reflective markers under different camera views. The pixel location of each marker is then extracted by using the Matlab built-in Circular Hough Transform (CHT) algorithm. This is shown on the left-hand side of Figure 6. A manual 2D-3D point correspondence is performed in order to associate each detected marker with its three-dimensional location in the M frame. At this stage, the Efficient Perspective-n-Points (EPnP) algorithm is used to solve the PnP problem and obtain an estimate of the roto-translation between the camera frame C and the object frame M. The estimation result is shown in the right-hand side of Figure 6. At this stage, an initial estimate of the constant roto-translation matrix $\boldsymbol{T}_C^I$ can be obtained from the roto-translations $\boldsymbol{T}_M^C$, $\boldsymbol{T}_O^M$ and $\boldsymbol{T}_O^I$, the latter two being returned by the VTS.

Subsequently, several pictures of the object M are taken with different camera views, and the CHT is applied to each of them to extract the pixel location of the retro-reflective markers. For each frame, the 2D-3D point correspondence can be made by using the initial estimate of



(a) (b)

Figure 6. Estimation of the roto-translation between the camera frame C and the markers frame M. The markers detection by the CHT algorithm (a) is shown beside the estimated roto-translation of the camera with respect to the the M object (b).

$\boldsymbol{T}_C^I$. The PnP problem can then be solved by means of a non-linear least squares solver, by minimizing the following sum of squares [28]:

$$\sigma_1(\boldsymbol{x} = \sum_{k=1}^{N_p}\sum_{j=1}^{N_m} \left\| \boldsymbol{p}_{f,i}(k) - \boldsymbol{\pi}\left(\boldsymbol{M}_{f,i}^O(k), \boldsymbol{T}_C^I, \boldsymbol{T}_O^I\right)\right\| \quad (4)$$

$$\boldsymbol{\pi}\left(\boldsymbol{M}_{f,i}^O(k), \boldsymbol{T}_C^I, \boldsymbol{T}_O^I\right) = \left(\frac{x_{f,i}^C}{z_{f,i}^C} f_x + c_x, \frac{y_{f,i}^C}{z_{f,i}^C} f_y + c_y\right) \quad (5)$$

$$\boldsymbol{M}_{f,i}^C = \begin{pmatrix} x_{f,i}^C & y_{f,i}^C & z_{f,i}^C \end{pmatrix}^T = \boldsymbol{R}_I^C \boldsymbol{R}_O^I \boldsymbol{M}_{f,i}^O + \boldsymbol{R}_I^C \boldsymbol{t}_O^I + \boldsymbol{t}_I^C \quad (6)$$

where N_m is the number of fiducial markers, N_a is the number of frames and $M_{f,i}^O$ represents the location of the ith marker in the marker frame M. The output of the minimization is a refined estimate of $\boldsymbol{T}_C^I$.

4.4. Estimation of the roto-translation matrix $\boldsymbol{T}_O^B$ - Mockup Calibration

The adopted procedure to estimate the roto-translation matrix $\boldsymbol{T}_O^B$ does not require the placement of retro-reflective markers on the Envisat mockup, taking advantage of the fact that the mockup is kept fixed throughout the image acquisition. The first step consists in acquiring a few images of the Envisat from different camera views (Figure 7). For each frame, the pixel location of pre-selected natural features of Envisat is hand-picked and a 2D-3D point correspondence is created with the three-dimensional points of an available 3D model. In this work, the corners of the Envisat body and of the SAR antenna were considered. Next, the EPnP is used to estimate the camera-to-Envisat roto-translation matrix $\boldsymbol{T}_B^C$ for each frame.

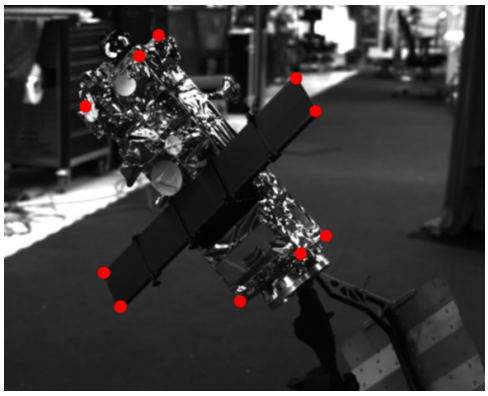


Figure 7. Example of a camera view of the Envisat mockup used for the estimation of the roto-translation matrix $\boldsymbol{T}_O^B$. The visible hand-picked corners are marked with red circles.

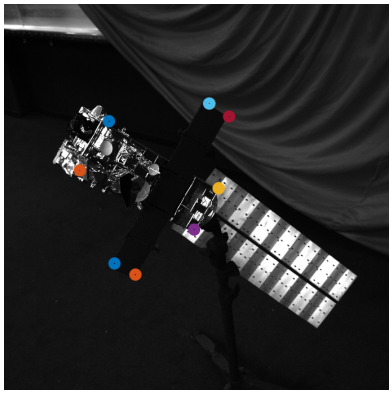


Figure 8. Reprojection of the hand-picked corners of Envisat with the refined estimate of $\boldsymbol{T}_O^B$ for a representative frames.

By knowing the roto-translation of the I frame with respect to the VICON origin O as well as the roto-translation matrix $\boldsymbol{T}_C^I$, it is then possible to obtain a raw estimate of the roto-translation matrix $\boldsymbol{T}_O^B$ for each frame. Due to the inaccuracies involved in the manual selection of the Envisat features as well as of the EPnP estimates of $\boldsymbol{T}_B^C$, the estimates of the constant matrix $\boldsymbol{T}_O^B$ will be different from each other and will require an additional refinement. This is accomplished once again by minimizing the total reprojection error of the selected features in a fashion similar to the one adopted in Section 4.3.1. Figure 8 shows the hand-picked features correctly reprojected with the refined estimate of $\boldsymbol{T}_O^B$ for a representative frame. Notably, and possibly due to inaccuracies still present in the estimates of $\boldsymbol{T}_C^I$ and $\boldsymbol{T}_O^B$, a larger reprojection error was observed for a few frames. As such, future adaptations of the calibration procedure, such as a more accurate calibration of both the mockup and the camera, should be considered to increase the validity of the reference pose labels for each generated image.

4.5. Estimation of the roto-translation matrix $\boldsymbol{T}_B^C$

Once the constant roto-translation matrices T_I^C and $\boldsymbol{T}_O^B$ are estimated, they can be used together with the VTS estimates of $\boldsymbol{T}_O^I$ to return the desired roto-translation $\boldsymbol{T}_B^C$ between the Envisat frame B and the camera frame C throughout the entire image acquisition trajectory.

5. CNN TRAINING AND TESTING

As already mentioned in Section 1, CNNs are currently emerging as a promising features extraction method. This is mostly due to the capability of their convolutional layers to extract high-level features of objects with improved robustness against image noise and illumination conditions [15]. Referring to Figure 9, the first essential step of keypoints-based CNN systems is represented by an Object Detection Network (e.g. Faster R-CNN [18], R-FCN [17] or MobileNet [6]) placed before the main CNN. The ODN regresses the coordinates of a bounding box around the target object, in order to crop a Region Of Interest (ROI) and to allow robustness to scale, variation, and background textures. The cropped ROI is then fed into a Keypoint Detection Network, which convolves with the input image and outputs a set of feature maps. These so-called *heatmaps* are detected around pre-selected features on the target object, such as corners or interest points. The 2D pixel coordinates of the heatmap's peak intensity characterize the predicted feature location, with the intensity and the shape indicating the confidence of locating the corresponding keypoint at this position [16]. Notably, the selection of the CNN will drive the achievable keypoints detection accuracy and robustness. Some architectures, such as the stacked Hourglass [12] and the U-Net [19], perform a downsampling of the input followed in series by an upsampling, in order to detect features at different scales. However, recent advancements in the field [2] demonstrated that by using parallel sub-networks across multiple resolutions, rather than multi-resolution serial stages, the CNN can manage to maintain a richer feature representation, facilitating more accurate and precise heatmaps. For this reason, the HRNet [26] architecture currently represents the state-of-the-art in keypoint detection, and it is chosen in the proposed pose estimation system.

5.1. Augmentation and Randomization Pipeline

Referring to Figure 10, the first step of the proposed pipeline for the datasets augmentation and randomization consists in generating ideal synthetic images of the Envisat 3D model. A highly-textured, realistic Envisat model is rendered in the Cinema 4D© software by keeping the virtual camera (Table 1) fixed and by randomly varying the pose of the rendering model with respect to the camera. Besides, the Azimuth and Elevation of the Sun are randomly varied by ± 40 deg around the

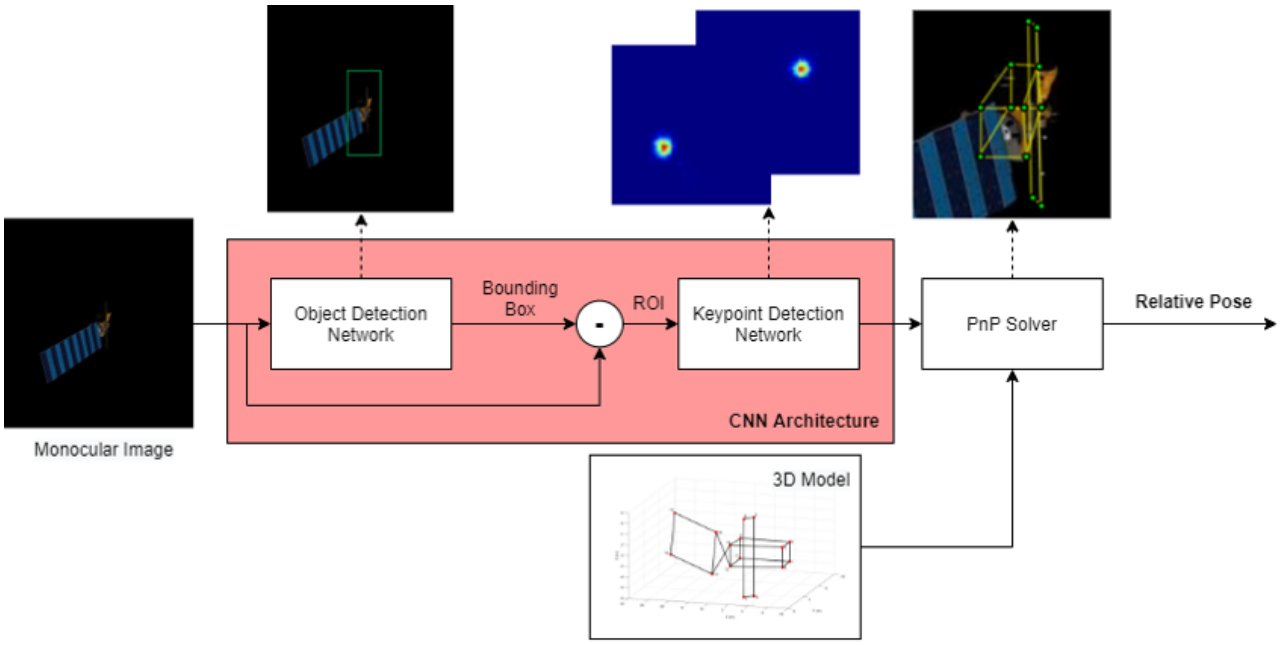


Figure 9. Proposed CNN architecture and interface with the PnP solver.

Table 1. Parameters of the camera used to generate the synthetic images in Cinema 4D©.

Parameter	Value	Unit
Image resolution	256×256	pixels
Focal length	$3.9{\cdot}10^{-3}$	m
Pixel size	$1.1{\cdot}10^{-5}$	m

ideal camera-Sun relative position, in order to recreate favourable as well as more adverse illumination conditions. Next, a randomization pipeline is introduced which adds the following effects to the rendering:

- Texture randomization. This is performed in order to increase the CNN robustness against texture variations between the synthetic and lab models of Envisat. The randomization is achieved in two different ways, by either adding a shader to each material in order to noise the textures, or by directly shuffling the textures of the materials.
- Light randomization. Four additional lights are introduced in random locations, aside from the main Sun illumination, in order to increase the CNN robustness against the illumination conditions recreated in the laboratory setup.
- Background randomization. Random scenes are used as image background in order to increase the CNN robustness against the laboratory environment. Specifically, external disturbance sources in the lab are likely to return non-zero pixel values in the image background, leading to inaccurate CNN detections in case the training dataset would consist of only black backgrounds.

Following the Cinema4D© rendering, an additional pipeline is used to further augment the generated images. This is performed by introducing the Earth in the background in some of the images and by corrupting the images with the following noise models:

- Gaussian, shot, impulse and speckle noise
- Gaussian, defocus, motion and zoom blurs
- Spatter, color jitter and random erase

Table 2 lists all the augmentation techniques together with the number of generated images. A total of 24,400 images were rendered and further split into training (70%), validation (15%) and test (15%) datasets.

5.2. Training, Validation and Test

During training, the validation dataset is used beside the training one to compute the validation losses and avoid overfitting. The Adam optimizer [7] is used with a cosine decaying learning rate with initial value of 10^{-3} and decaying factor of 0.1. The network is trained for a total of 210 epochs. Finally, the network performance after training is assessed with the synthetic test dataset.

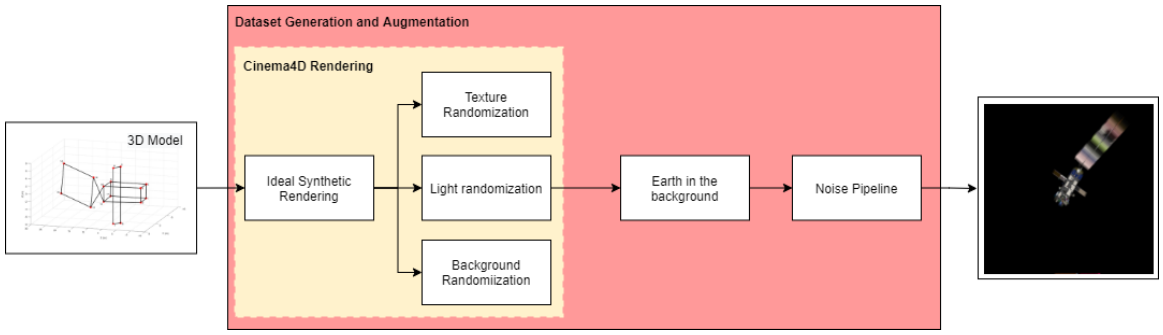
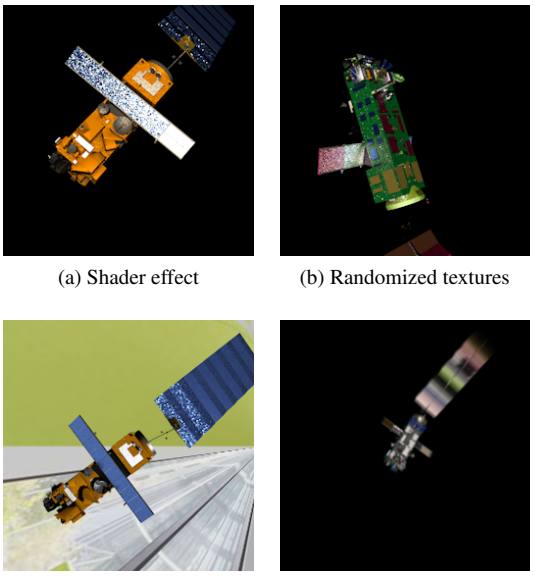


Figure 10. Dataset Augmentation Pipeline.

Table 2. Augmentation Breakdown.

Description	number of Images
No augmentations	1000
Random lights	550
Random lights & textures	2000
Random lights & background	350
Randomization & Noise & Earth	20,500
Total	**24,400**



(a) Shader effect (b) Randomized textures

(c) Random background (d) Motion blur

Figure 11. Output examples of the randomization pipeline.

The performance is assessed in terms of Root Mean Squared Error (RMSE) between the ground truth (GT) and the x,y coordinates of the extracted features, which is computed as

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{n_{tot}} [(x_{GT,i} - x_i)^2 + (y_{GT,i} - y_i)^2]}{n_{tot}}}. \quad (7)$$

The CNN performance on the test dataset shows a mean detection accuracy of 0.97, with a RMSE mean $\mu = 2.78$ pxl and a Mean Absolute Deviation (MAD) of 2.87 pxl. Overall, this proves that the network is capable of accurately detecting the pre-trained keypoint features in most of the test images. Figure 12 shows a mosaic of keypoint detection results on a subset of the test dataset. Notably, wrong detections occur when the solar panel completely hides the main Envisat body. However, the CNN returns good detection accuracies when only parts of Envisat are occluded, demonstrating the capability of learning the relative position between features during partial observability (Figure 13).

6. POSE ESTIMATION

Following the promising results presented by the authors in [15], the CEPPnP method [4] is selected to estimate the relative pose from the detected features. In this method, the CNN heatmaps around the detected features are exploited to derive feature covariance matrices and capture the statistical distribution of the detected features.

The first step of the CEPPnP algorithm is to rewrite the PnP problem in Eqs.1-2 as a function of a 12-dimensional vector $\boldsymbol{y}$ containing the control point coordinates in the camera reference system,

$$\boldsymbol{M}\boldsymbol{y} = \boldsymbol{0}, \quad (8)$$

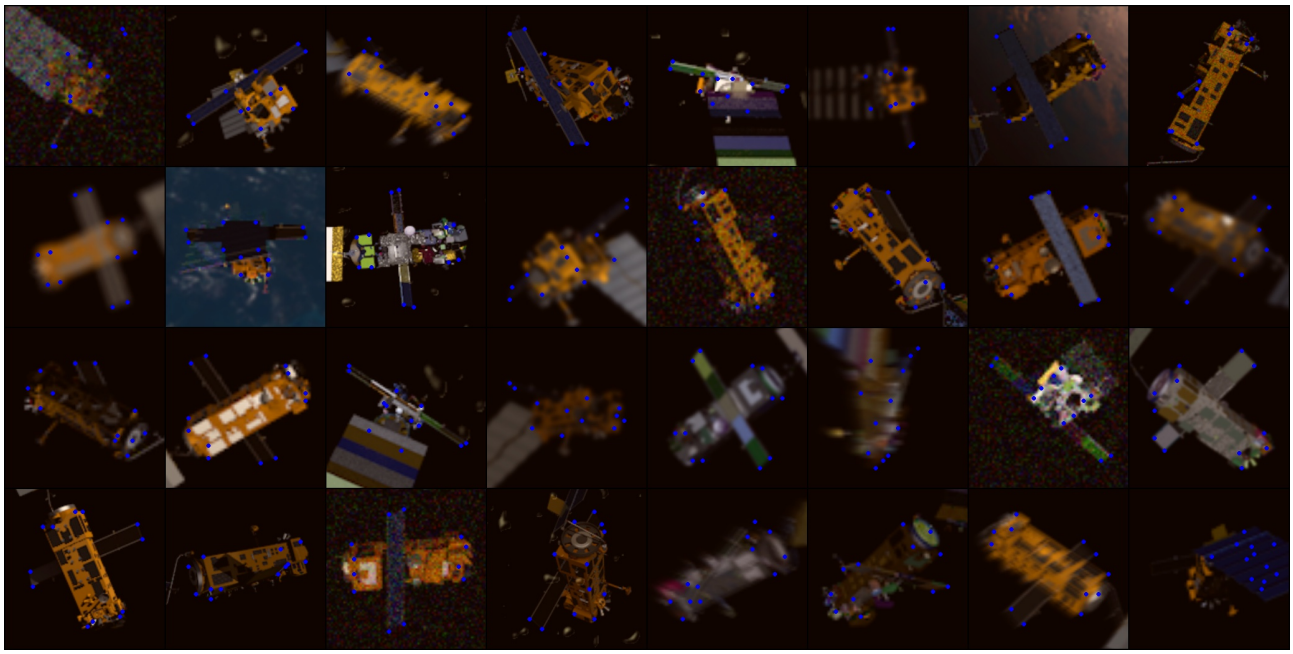
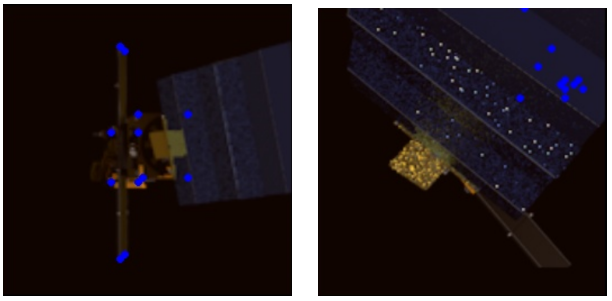


Figure 12. Mosaic of keypoints detection results on a subset of the test dataset.



(a) acc = 0.95, RMSE = 0.47 pxl (b) RMSE =93 pxl

Figure 13. Example of high (a) and low (b) detection accuracies during poor visibility or occlusion.

where $\boldsymbol{M}$ is a $2n\times12$ known matrix. This is the fundamental equation in the EPnP problem [10]. The likelihood of each observed feature location $\boldsymbol{u}_i$ is then represented as

$$P(\boldsymbol{u}_i) = k \cdot e^{-\frac{1}{2}\Delta\boldsymbol{u}_i^T \boldsymbol{C}_{\boldsymbol{u}_i}^{-1}\Delta\boldsymbol{u}_i}, \tag{9}$$

where $\Delta\boldsymbol{u}_i$ is a small, independent and unbiased noise with expectation $E[\Delta\boldsymbol{u}_i] = \boldsymbol{0}$ and covariance $E[\Delta\boldsymbol{u}_i\Delta\boldsymbol{u}_i^T] = \sigma^2\boldsymbol{C}_{\boldsymbol{u}_i}$ and k is a normalization constant. Here, σ^2 represents the global uncertainty in the image, whereas $\boldsymbol{C}_{\boldsymbol{u}_i}$ is the 2x2 unnormalized covariance matrix representing the Gaussian distribution of each detected feature, computed from the CNN heatmaps. After some calculations [4], the EPnP formulation can be rewritten as

$$(\boldsymbol{N} - \boldsymbol{L})\boldsymbol{y} = \lambda\boldsymbol{y}. \tag{10}$$

This is an eigenvalue problem in which both $\boldsymbol{N}$ and $\boldsymbol{L}$ matrices are a function of $\boldsymbol{y}$ and $\boldsymbol{C}_{\boldsymbol{u}_i}$. The problem is solved iteratively by means of the closed-loop EPPnP solution for the four control points, assuming no feature uncertainty. Once $\boldsymbol{y}$ is estimated, the relative pose is computed by solving the generalized Orthogonal Procrustes problem used in the EPPnP [5].

To derive $\boldsymbol{C}_{\boldsymbol{u}_i}$ for each feature, each heatmap distribution is used to compute a weighted covariance between x, y,

$$\boldsymbol{C}_{\boldsymbol{u}_i} = \begin{pmatrix} \mathrm{cov}(x,x) & \mathrm{cov}(x,y) \\ \mathrm{cov}(y,x) & \mathrm{cov}(y,y) \end{pmatrix}, \tag{11}$$

where

$$\mathrm{cov}(x,y) = \sum_{i=1}^{n} w_i(x_i - p_x)\cdot(y_i - p_y) \tag{12}$$

and n is the number of pixels in each feature's heatmap. In order to represent a distribution around the peak of the detected feature, rather than around the heatmap's mean, the mean is replaced by the peak location $\boldsymbol{p} = (p_x, p_y)$. This is particularly relevant when the heatmaps are asymmetric and their mean does not coincide with their peak.

7. RESULTS

In this section, the pose estimation results are presented for both the synthetic test dataset and the mockup images generated at the ORGL facility of ESTEC. Two separate

error metrics are adopted in the evaluation, in accordance with Kisantal et al. [8]. Firstly, the translational error between the estimated relative position $\hat{\boldsymbol{t}}^C$ and the ground truth $\boldsymbol{t}$ is computed as

$$E_T = \left\| \boldsymbol{t}^C - \hat{\boldsymbol{t}}^C \right\|. \tag{13}$$

This metric is also applied for the translational and rotational velocities estimated in the navigation filter. Secondly, the attitude accuracy is measured in terms of the Euler axis-angle error between the estimated quaternion $\hat{\boldsymbol{q}}$ and the ground truth $\boldsymbol{q}$,

$$\boldsymbol{\beta} = (\beta_s \quad \boldsymbol{\beta_v}) = \boldsymbol{q} \otimes \hat{\boldsymbol{q}} \tag{14}$$

$$E_R = 2 \arccos\left(|\beta_s|\right). \tag{15}$$

7.1. Synthetic Test Dataset

The CNN-detected keypoints and the heatmaps-derived covariances are fed into the CEPPnP solver together with the intrinsic camera parameters and the Envisat 3D model to solve for the relative pose. Besides, the performance of the CEPPnP algorithm is evaluated against the covariance-free EPnP solver [10], to assess the impact of the feature covariance on the pose estimation accuracy. Table 3 shows the results across the test dataset in terms of mean μ and MAD.

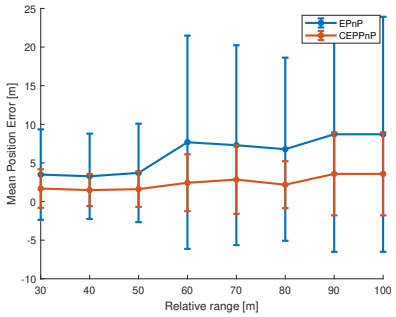
Table 3. Pose Estimation performance results for the synthetic test, expressed in terms of $\mu\pm$ MAD

Metric	CEPPnP	EPPnP
E_T [m]	2.25 ± 3.3	5.83 ± 10.2
E_R [deg]	2.7 ± 2.8	2.3 ± 2.6

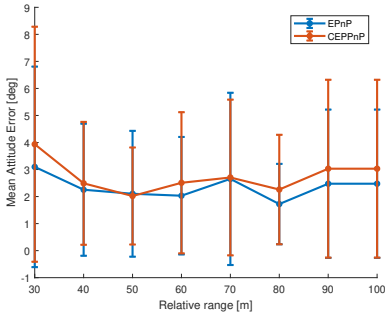
Additionally, the estimated relative position and attitude are expressed as a function of the relative range in order to capture the trend of the pose estimation accuracy for increasing relative distances. This is represented in Figure 14. As can be seen, including the heatmaps-derived covariances results in a more robust and accurate pose estimation, specifically due to an improved estimate of the relative position (Figure 14a). In other words, the CEPPnP position estimate is characterized by a more accurate mean μ and a smaller MAD, indicating that the estimation performance is improved in those scenarios for which the CNN detections are less accurate. These considerations confirm the results reported by the authors in earlier works [15].

7.2. ORGL Dataset

The CNN performance on the ORGL dataset is evaluated at both keypoints detection and pose estimation levels. Firstly, the keypoints detection of the proposed CNN,



(a)

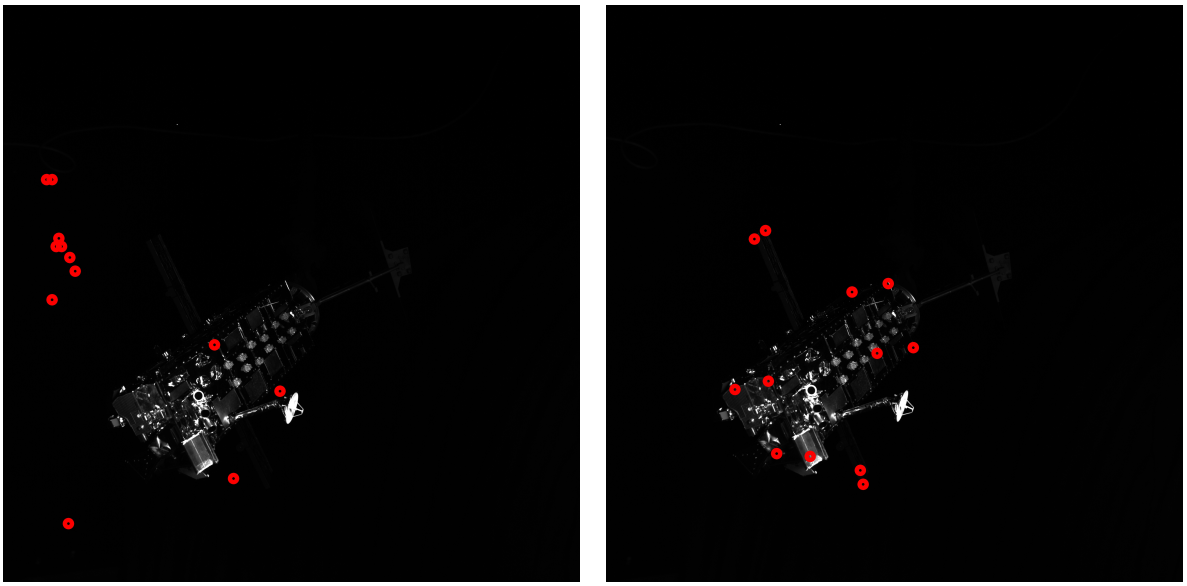


(b)

Figure 14. Pose Estimation Results - The standard deviation of the position (a) and attitude (b) errors is depicted as the length of each error bar above and below the mean errors E_T,E_R.

trained with the randomized training dataset, is compared with the keypoints detection of the same CNN trained on a subset of the augmented dataset, characterized only by Earth in the background and noise. This is shown in Figure 15 for a sample image. Due to a lack of background, light and texture randomization in the training dataset, the CNN trained only on the partially-augmented dataset is overfitted on the textures learned on the synthetic Envisat model. As a result, the network cannot associate the correct texture to each feature, and the detected keypoints are randomly scattered around the image (Figure 15a). Conversely, the CNN trained on the randomized dataset proves to be more robust against variations in texture and light between the synthetic and the lab images, inferring the correct shape of the mockup and detecting most of the keypoints in the correct location (Figure 15b). This improved robustness is mostly linked to the capability of the CNN to learn shapes rather than textures, which can be traced back to the textures randomization step included in the augmentation and randomization pipeline. Remarkably, the features are detected even without a high synthetic-lab representativeness, showing the CNN capability to transfer to images which considerably differ from the training ones.

Next, the pose estimates are compared with the reference pose labels computed from the calibration procedure described in Section 4. Table 4 lists the pose estimation error across 100 ORGL images. As can be seen, the pose estimates result in a large mean attitude error, despite an



(a) No randomization

(b) Light/textures/background randomization

Figure 15. Impact of light, textures and background randomization on the CNN detection performance for a sample ORGL image. Notably, the randomization of the training dataset improves the CNN robustness against different light, texture and background conditions.

Table 4. Pose Estimation performance results for the ORGL images, expressed in terms of $\mu\pm$ MAD

Metric	CEPPnP	EPPnP
E_T [m]	0.77 ± 0.68	0.78 ± 0.99
E_R [deg]	30.5 ± 22.7	20.8 ± 13.7

accurate estimation of the relative position. Moreover, the inclusion of feature covariances in the PnP solver does not seem to improve the estimation accuracy. There are at least two potential causes of this behaviour. On the one hand, the relative distance between the monocular camera and the Envisat mockup is of approximately 1 m, meaning that relatively small pixel errors can lead to large attitude errors. On the other hand, slight differences between the shape of the rendering model and the mockup could affect the attitude estimate more than the position. Besides, the reference attitude labels could be inaccurate for some of the generated images, mostly due to some inaccurate VTS telemetry or as a result of inaccurate estimates of the roto-translation matrices in Section 4.3.1.

Nevertheless, pose estimation results for a subset of the ORGL dataset (including Figure 15b) are characterized by relative attitude errors of $<$4 deg and relative position errors of 5 cm, even in presence of adverse illumination conditions. This shows not only that accurate pose estimates can be returned by the proposed CNN-based pose estimation system, but also that most of the reference pose labels can be used to validate the performance of the system on lab-generated images.

8. CONCLUSIONS AND RECOMMENDATIONS

This paper introduces a framework for the on-ground validation of a CNN-based monocular pose estimation system for uncooperative spacecraft. A calibration procedure is proposed to support the generation of realistic laboratory images of a 1:25 scaled mockup of the Envisat spacecraft. These images are used to test the capability of the proposed CNN to bridge the gap between synthetic training and laboratory testing whilst returning accurate pose estimates.

The adopted CNN is validated at different levels of the proposed pose estimation system, by assessing its performance both in terms of keypoints detection and pose estimate. At a keypoint detection level, the system proves to benefit from the augmentation and randomization of the dataset used during the CNN training. The results show that the robustness of the CNN against lab-generated images can be increased by randomizing lights, material textures and image background. This also allows to increase the robustness against differences between the rendering model and the mockup. At a pose estimation level, the results on the synthetic test dataset indicate that the covariant-based CEPPnP solver returns more accurate pose estimates than the standard EPnP solver, thanks to the capability of the heatmaps-based covariances to capture the statistical information of the detected features. Besides, the accurate pose estimates reported for a subset of the ORGL dataset indicate that the system built on a synthetic training can transfer to more realistic images of the target. Furthermore, it demonstrates that the calibration procedure returns accurate pose labels for most

of the generated images.

However, further work is still required. First of all, the calibration procedure shall be revisited and improved in order to guarantee an accurate and reliable pose label for each generated image. Secondly, different augmentation and randomization pipelines shall be investigated in order to further improve the CNN performance. Finally, a more comprehensive dataset of lab-generated images shall be created in order to address the impact of including realistic images on the training on the CNN performance.

ACKNOWLEDGMENTS

This study is funded and supported by the European Space Agency and Airbus Defence and Space under Network/Partnering Initiative (NPI) program with grant number NPI 577 - 2017. The first author would like to thank Martin Schwendener and Irene Huertas for the help during the image acquisition campaign at ORGL, and Kuldeep Barad for the adaptation of the HRNet.

REFERENCES

1. K. Black, S. Shankar, D. Fonseka, J. Deutsch, A. Dhir, and M. Akella. Real-time, flight-ready, non-cooperative spacecraft pose estimation using monocular imagery. In *31st AAS/AIAA Space Flight Mechanics Meeting*, 2021.
2. B. Chen, J. Cao, A. Parra, and T. Chin. Satellite Pose Estimation with Deep Landmark Regression and Nonlinear Pose Refinement. In *International Conference on Computer Vision*, Seoul, South Korea, 2019.
3. S. D'Amico, M. Benn, and J. Jorgensen. Pose Estimation of an Uncooperative Spacecraft from Actual Space Imagery. *International Journal of Space Science and Engineering*, 2(2):171–189, 2014.
4. L. Ferraz, X. Binefa, and F. Moreno-Noguer. Leveraging Feature Uncertainty in the PnP Problem. In *Proceedings of the British Machine Vision Conference*, Nottingham, UK, 2014.
5. L. Ferraz, X. Binefa, and F. Moreno-Noguer. Very Fast Solution to the PnP Problem with Algebraic Outlier Rejection. In *IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, USA, 2014.
6. A.G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. In *ArXiv Preprint*, 2017.
7. D.P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *3rd International Conference for Learning Representations*, San Diego, CA, USA, 2015.
8. M. Kisantal, S. Sharma, T.H. Park, D. Izzo, M. Martens, and S. D'Amico. Satellite Pose Estimation Challenge: Dataset, Competition Design and Results. *IEEE Transactions on Aerospace and Electronic Systems*, 2020.
9. H. Krúger and S. Theil. TRON - hardware-in-the-loop test facility for lunar descent and landing optical navigation. In *IFAC-ACA 2010 Automatic Control in Aerospace*, 2010.
10. Lepetit, F. Moreno-Noguer, and P. Fua. EPnP: an accurate O(n) solution to the PnP problem. *International Journal of Computer Vision*, 81:155–166, 2009.
11. P. Merriaux, Y. Dupuis, R. Boutteau, P. Vasseur, and X. Savatier. A study of vicon system positioning performance. *Sensors*, 17(7):1591, 2017.
12. A. Newell, K. Yang, and J. Deng. Stacked hourglass networks for human pose estimation. In B. Leibe, J. Matas, N. Sebe, and M. Welling, editors, *Computer Vision - ECCV 2016*, volume 9912, pages 483–499. Springer, Cham, 2016.
13. T.H. Park, S. Sharma, and S. D'Amico. Towards Robust Learning-Based Pose Estimation of Noncooperative Spacecraft. In *AAS/AIAA Astrodynamics Specialist Conference*, Portland, ME, USA, 2019.
14. L. Pasqualetto Cassinis, R. Fonod, and E. Gill. Review of the Robustness and Applicability of Monocular Pose Estimation Systems for Relative Navigation with an Uncooperative Spacecraft. *Progress in Aerospace Sciences*, 110, 2019.
15. L. Pasqualetto Cassinis, R. Fonod, and E. Gill. Evaluation of tightly- and loosely-coupled approaches in CNN-based pose estimation systems for uncooperative spacecraft. *Acta Astronautica*, 182:189–202, 2021.
16. G. Pavlakos, X. Zhou, A. Chan, K.G. Derpanis, and K. Daniilidis. 6-DoF Object Pose from Semantic Keypoints. In *IEEE International Conference on Robotics and Automation*, 2017.
17. S. Ren, K. He, R. Girshick, and J. Sun. Object detection via region-based fully convolutional networks. *Advances in neural information processing systems*, pages 379–387, 2016.
18. S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1137 – 1149, 2017.
19. O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention*, pages 234–241. Springer, 2015.
20. S. Sharma, C. Beierle, and S. D'Amico. Pose Estimation for Non-Cooperative Spacecraft Rendezvous using Convolutional Neural Networks. In *IEEE Aerospace Conference*, Big Sky, MT, USA, 2018.

21. S. Sharma and S. D'Amico. Comparative Assessment of Techniques for Initial Pose Estimation Using Monocular Vision. *Acta Astronautica*, 123:435–445, 2015.

22. S. Sharma and S. D'Amico. Pose Estimation for Non-Cooperative Spacecraft Rendezvous using Neural Networks. In *29th AAS/AIAA Space Flight Mechanics Meeting*, Ka'anapali, HI, USA, 2019.

23. S. Sharma, J. Ventura, and S. D'Amico. Robust Model-Based Monocular Pose Initialization for Noncooperative Spacecraft Rendezvous. *Journal of Spacecraft and Rockets*, 55(6):1–16, 2018.

24. J.F. Shi, S. Ulrich, and S. Ruel. CubeSat Simulation and Detection using Monocular Camera Images and Convolutional Neural Networks. In *2018 AIAA Guidance, Navigation, and Control Conference*, Kissimmee, FL, USA, 2018.

25. S. Sonawani, R. Alimo, R. Detry, D. Jeong, A. Hess, and H. Ben Amor. Assistive relative pose estimation for on-orbit assembly using convolutional neural networks. In *AIAA Scitech 2020 Forum*, Orlando, FL, USA, 2020.

26. K. Sun, B. Xiao, D. Liu, and J. Wang. Deep high-resolution representation learning for human pose estimation. In *2019 IEEE Conference on Computer Vision and Pattern Recognition*, Long Beach, CA, USA, 2019.

27. A. Tatsch, N. Fitz-Coy, and S. Gladun. On-orbit Servicing: A brief survey. In *Proceedings of the 2006 Performance Metrics for Intelligent Systems Workshop*, pages 21–23, 2006.

28. A. Valmorbida, M. Mazzucato, and M. Pertile. Calibration procedures of a vision-based system for relative motion estimation between satellites flying in proximity. *Measurement*, 151, 2020.

29. M. Wieser, H. Richard, G. Hausmann, J-C. Meyer, S. Jaekel, M. Lavagna, and R. Biesbroek. e.deorbit mission: OHB debris removal concepts. In *ASTRA 2015-13th Symposium on Advanced Space Technologies in Robotics and Automation*, Noordwijk, The Netherlands, 2015.

30. M. Wilde, C. Clark, and M. Romano. Historical survey of kinematic and dynamic spacecraft simulators for laboratory experimentation of on-orbit proximity maneuvers. *Progress in Aerospace Sciences*, 110, 2019.

31. M. Zwick, I. Huertas, L. Gerdes, and G. Ortega. ORGL - ESA's test facility for approach and contact operations in orbital and planetary environments. In *International Symposium on Artificial Intelligence, Robotics and Automation in Space*, Madrid, Spain, 2018.