# CubeSat-CDT: A Cross-Domain Dataset for 6-DoF Trajectory Estimation of a Symmetric Spacecraft

Mohamed Adel Musallam[1 [0000−0002−2245−6629]], Arunkumar Rathinam[1 [0000−0002−2231−0734]], Vincent Gaudillière[1 [0000−0002−6498−7823]], Miguel Ortiz del Castillo[1 [0000−0001−8431−3765]], and Djamila Aouada[1 [0000−0002−7576−2064]]

SnT, University of Luxembourg, L-1359 Luxembourg
https://cvi2.uni.lu/
mohamed.ali@uni.lu, arunkumar.rathinam@uni.lu,
vincent.gaudilliere@uni.lu, djamila.aouada@uni.lu

**Abstract.** This paper introduces a new cross-domain dataset, *CubeSat-CDT*, that includes 21 trajectories of a real CubeSat acquired in a laboratory setup, combined with 65 trajectories generated using two rendering engines – *i.e.* Unity and Blender. The three data sources incorporate the same 1U CubeSat and share the same camera intrinsic parameters. In addition, we conduct experiments to show the characteristics of the dataset using a novel and efficient spacecraft trajectory estimation method, that leverages the information provided from the three data domains. Given a video input of a target spacecraft, the proposed end-to-end approach relies on a Temporal Convolutional Network that enforces the inter-frame coherence of the estimated 6-Degree-of-Freedom spacecraft poses. The pipeline is decomposed into two stages; first, spatial features are extracted from each frame in parallel; second, these features are lifted to the space of camera poses while preserving temporal information. Our results highlight the importance of addressing the domain gap problem to propose reliable solutions for close-range autonomous relative navigation between spacecrafts. Since the nature of the data used during training impacts directly the performance of the final solution, the *CubeSat-CDT* dataset is provided to advance research into this direction.

## 1 Introduction

With the increase in the number of space missions and debris [17,3,9], the need for Space Situational Awareness (SSA) – referring to the key ability of inferring reliable information about surrounding space objects from embedded sensors – is growing rapidly. Moreover, the highest level of autonomy is required to meet

This work was funded by the Luxembourg National Research Fund (FNR), under the project reference BRIDGES2020/IS/14755859/MEET-A/Aouada, and by LMO (https://www.lmo.space).

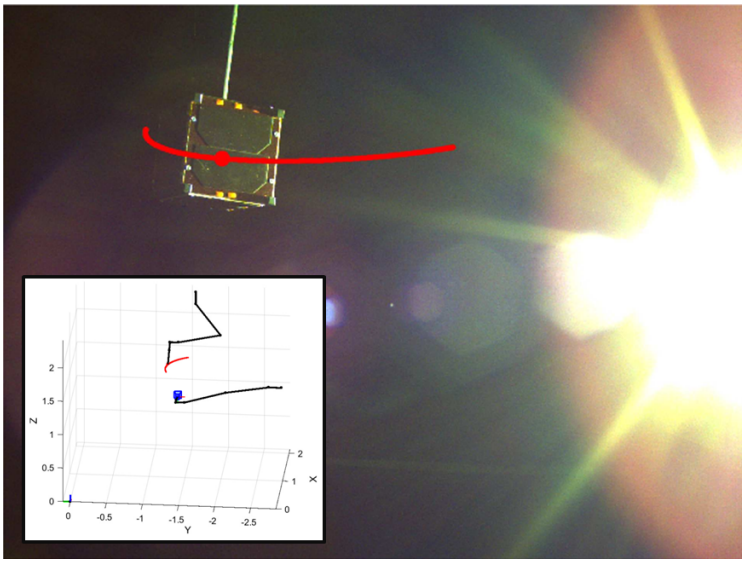



Fig. 1: Illustration of the SnT Zero-G Laboratory data. Top: chaser camera field of view featuring a CubeSat target with its projected trajectory (in red). Bottom left: instantaneous 3D positions of the robotic arms simulating the chaser (camera in blue) and target (trajectory in red) spacecrafts.

the need for reactivity and adaptation during on-orbit operations. Due to their low cost and power consumption combined with their high frame rate, cameras represent suitable sensors for SSA. Consequently, vision-based navigation is the preferred route for performing autonomous in-orbit operations around a target spacecraft [24]. To do so, the core task consists in estimating both the position and attitude – referred to as pose – of the target over time. Furthermore, Deep Learning (DL) techniques have been proven to be successful in a wide variety of visual applications such as image classification, object detection or semantic segmentation [6]. Therefore, their use in monocular spacecraft pose estimation has gained interest accordingly. Moreover, the results from the first edition of the Satellite Pose Estimation Challenge (SPEC) [15] – organized by the Advanced Concepts Team (ACT) of the European Space Agency and the Space Rendezvous Laboratory (SLAB) of Stanford University – have shown promising outcomes in that direction. The scope of the SPEC was limited to single-frame pose estimation from synthetic images only.

Due to the appearance gap between images from synthetic and real domains, DL-based algorithms trained on synthetic data typically suffer from significant performance drop when tested on real images [15]. As a consequence, existing spaceborne images captured from previous missions are sometimes combined with synthetic data. In particular, the Cygnus dataset [4] contains 540 pictures of the Cygnus spacecraft in orbit in conjunction with 20k synthetic images generated with Blender [1]. However, the main limitation of spaceborne images is

| | SPARK [20] | SPEED [15] | SPEED+ [21] | URSO [26] | SwissCube [13] | Cygnus [4] | Prisma12K [22] | Prisma25 [7] | **CubeSat-CDT** |
|---|---|---|---|---|---|---|---|---|---|
| Synthetic images | 150k | 15k | 60k | 15k | 50k | 20k | 12k | - | **14k** |
| Non-synthetic images | - | 305 | 10k | - | - | 540 | - | 25 | **8k** |
| Object classes | 15 | 1 | 1 | 2 | 1 | 1 | 1 | 1 | **1** |
| Image Resolution | 1024 × 1024 | 1920 × 1200 | 1920 × 1200 | 1080 × 960 | 1024 × 1024 | 1024 × 1024 | 752 × 580 | -NA- | **1440 × 1080** |
| Visible | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Color | ✓ | ✗ | ✗ | ✓ | ✓ | ✓ | ✗ | ✗ | ✓ |
| Depth | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ |
| Mask | ✓ | ✗ | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ |
| 6D Pose | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Rendering | sim | sim + lab | sim + lab | sim | sim | sim + real | sim | real | **sim + lab** |
| Trajectories | ✗ | ✗ | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✓ |
| Public dataset | ✓ | ✓ | ✓ | ✓ | ✓ | ✗ | ✗ | ✗ | ✓ |

Table 1: Overview of existing SSA datasets. In the table, *sim* refers to simulated images and *lab* to laboratory data.

the lack of accurate pose labels and their limited diversity in terms of pose distribution. To overcome these difficulties, laboratory setups trying to mimic space conditions currently represent the de facto target domain for spacecraft pose estimation algorithms [21]. Moreover, laboratories offer monitoring mockup poses and environmental conditions to ensure a higher quality of the data [30,23]. Therefore, the ongoing second edition of the SPEC will rank the pose estimation algorithms based on their performance on two laboratory datasets, while training images were generated synthetically. That so-called SPEED+ dataset [21] is the first of its kind for vision-only spacecraft pose estimation that combines the information from 60k synthetic images with 10k others acquired from a robotic laboratory setup. However, laboratory pose labels are not publicly available.

The SPEED+ dataset offers no temporal consistency between the images. While single-frame spacecraft pose estimation is needed to initialize any multi-frame tracking algorithm [16,28], leveraging the temporal information provided through consecutive image acquisitions is likely to improve the robustness and reliability of any deployable method. However, to the best of our knowledge, no existing dataset features temporally consistent image sequences for 6-DoF spacecraft trajectory estimation.

In this paper, we propose a novel dataset with the aim to foster research on both the domain gap reduction and temporal information processing for spacecraft pose estimation. To the best of our knowledge, this is the first dataset featuring multiple (synthetic-2x and real-1x) domains with temporal information for spacecraft 6-DoF trajectory estimation (86 trajectories). We also propose a baseline algorithm that leverages the data from the three different modalities contained in the dataset.

In addition, man-made objects and specially spacecrafts often present a high degree of symmetry by construction. However, most existing datasets rely on non-symmetrical target spacecrafts [20,8,15,21,26,13,4,22,7]. Indeed, estimating the pose – especially orientation – of a symmetric object is a difficult task that has been receiving interest from the research community [5,25,12]. To develop it further, our dataset features a highly-symmetrical spacecraft – a 1U CubeSat. Moreover, instead of a mockup made with inadequate materials, a real space-compliant spacecraft is used.

Our contributions are summarized below:

- A comparative analysis of the existing SSA datasets.
- A cross-domain spacecraft trajectory dataset, that will be made publicly available and referred to as *CubeSat-CDT*, where "CDT" stands for Cross-Domain Trajectories.
- A novel algorithm for spacecraft trajectory estimation, built upon previous work [19], whose training has been made possible by the creation of the CubeSat-CDT dataset.
- A detailed analysis of the proposed dataset based on the training and testing of the aforementioned algorithm.

The paper is organised as follows. Section 2 provides a comparative review of existing SSA datasets. Section 3 describes the proposed CubeSat-CDT dataset. Section 4 presents the proposed method to leverage the temporal information for trajectory estimation and cross domain data validation. Section 5 presents an analysis of the performance of our proposed approach on the CubeSat-CDT dataset.

## 2 Related datasets

Multiple datasets are provided to address the SSA challenge. Some are freely available for research [20,15,21,26,13], others are proprietary from some space companies and are not publicly available [4,22,7]. These datasets are discussed below, highlighting their contributions, limitations and difference with CubeSat-CDT. Table 1 summarizes this study.

SPEED [15] consists of 15k grayscale images of the Tango spacecraft, with a resolution of 1920×1200 pixels. Two different sets of images are available: a) 300 generated in a robotic laboratory environment with a custom illumination set up to recreate the space environment; b) 15k images of the exact mock-up generated using Open-GL to generate photo-realistic images. The main limitation of this study is the unbalanced number of samples from the laboratory and the synthetic data, as well as the absence of pose labels for most of the laboratory data.

SPEED+ [21] introduces the first dataset designed to minimize the gap between synthetic simulated and real data. In addition to the 60k synthetic generated images of the Tango satellite using Open-GL, it incorporates 10k laboratory images acquired with two different sources of illumination: *lightbox*, with diffuser plates for albedo simulation, and *sunlamp* to mimic direct high-intensity homogeneous light from the Sun.

SPARK [20] is a synthetic dataset generated using the Unity3D game engine as a simulation environment. It has 150k images of 10 different satellite models along with 5 debris objects combined in one class. It contains a large diversity in sensing conditions for all the targets, and provides RGB and depth information. Nonetheless, SPARK, intended to be used for spacecraft recognition and pose estimation, does not include any trajectory.

URSO [26] provides 15k images of two different targets, the 'Dragon' spacecraft and the 'Soyuz' one, with different operating ranges and at a resolution

| Data Domain \ Distance | Minimum | Maximum |
|---|---|---|
| Zero-G Lab | 0.65m | 1.2m |
| Synthetic (SPARK) | 0.85m | 3.8m |
| Synthetic (Blender) | 0.40m | 2.2m |

Table 2: Minimum and maximum distances between the CubeSat and the camera for each data domain of the proposed CubeSat CDT dataset.

of 1080×960 pixels. The images were randomly generated and sampled around the day side of the Earth from low Earth orbit altitude with an operating range between 10m and 40m. All images are labelled with the corresponding target pose with respect to the virtual vision sensor.

SwissCube [13] is made of 500 scenes consisting of 100 frame sequences for a total of 50k images. It is the first public dataset that includes spacecraft trajectories, in particular a 1U CubeSat. Nonetheless, the main limitation of the dataset is the domain, as it only contains synthetically generated images using Mitsuba 2 render.

Cygnus [4] includes 20k synthetic images generated with Blender in addition to 540 real images of the Northrop Grumman Enhanced Cygnus spacecraft. They perform several augmentation techniques on the synthetic data including various types of randomized glare, lens flares, blur, and background images. However, the main limitation of the dataset is that it is not publicly accessible.

PRISMA12K [22] is created using the same rendering software used in SPEED. However, PRISMA12K replicates the camera parameters used during the PRISMA mission targeting the Mango satellite. It comprises 12k grayscale images of the Tango spacecraft using the same pose distribution presented in SPEED.

PRISMA25 [7] contains 25 spaceborne images captured during the rendezvous phase of the PRISMA mission. This real dataset is used to evaluate the performance of the algorithms developed using PRIMSA12K. The main limitation of this dataset is the number of real case examples and the lack of diversity in the target's pose.

The CubeSat-CDT dataset, presented in Section 3, contains multiple trajectories in three different domains with real spacecraft (see Table 1). We believe such a dataset opens new possibilities for studying trajectory estimation of spacecrafts.

## 3 Proposed CubeSat-CDT dataset

The proposed CubeSat Cross-Domain Trajectory (CDT) dataset contains 21 trajectories of a real CubeSat acquired in a laboratory setup, combined with 50 trajectories generated using Unity [10], and 15 trajectories generated using Blender [1]. Combining a total of 22k high-quality and high-resolution images of a 1U CubeSat moving in predefined trajectories. Table 2 shows the minimum

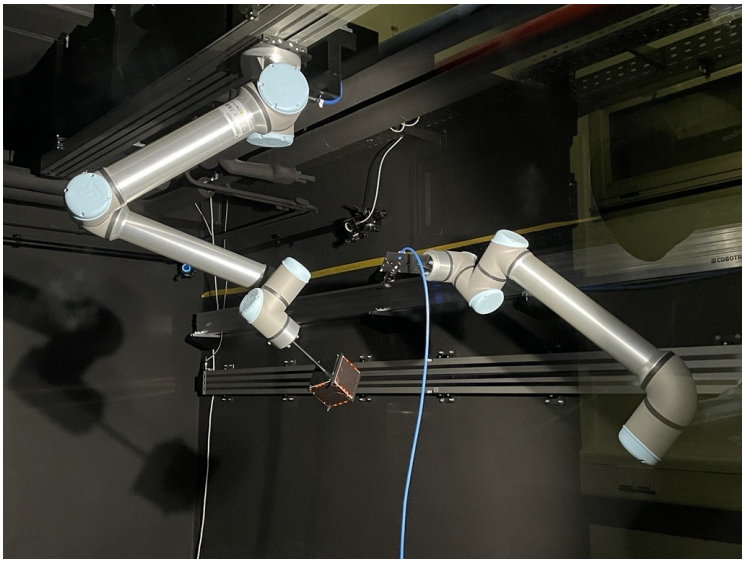


Fig. 2: Illustration of the Zero-Gravity Laboratory, located at the Interdisciplinary Centre for Security, Reliability and Trust (SnT) of the University of Luxembourg [23].

and maximum distances between the CubeSat and the camera according to the data domain.

### 3.1 Zero-G Laboratory Data

For this work we utilized the Zero-Gravity facility (Zero-G Lab) at the Interdisciplinary Center for Security, Reliability and Trust (SnT) of the University of Luxembourg [23]. In this facility we reconstruct the space environment replicating multiple environmental conditions from the illumination to the motion in space. The Zero-G Lab, is a 3x5 meters robotic based laboratory used for real-time simulation of on-orbit servicing missions. It consists of two industrial, highly accurate robotic arms with six degrees of freedom each from Universal Robots (model UR10) [27]. They can simulate the 6D dynamic motion of two satellites during orbital rendezvous. Both arms are mounted on top of a Cobotrack rail system with a maximum translational velocity of 1.45m/s. One robot is mounted on the wall at a linear track of 5m length and the other one is mounted on the ceiling at a linear track of 4.6m length. On both tracks the robotic manipulators can be moved and thus the final approach can be completely simulated (see Fig. 2).

The camera mounted as a payload of the wall robotic arm is the FLIR Blackfly S BFS-U3-16S2C. This camera is a lightweight (<50g) and cost effective solution for space-sensitive imaging applications. This camera has a variety of

features, including precise control over exposure, gain, white balance, and color correction. A fixed focal lens of 12mm, designed for pixels that are $\leq 2.2\mu$m, was added to the camera. This provides a high level resolution (>200 lp/mm) across the sensor.

### 3.2 Unity-based SPARK Synthetic Data

The first subset of synthetic data, referred to in the following as *SPARK Synthetic Data*, was generated using the SPARK space simulation environment based on Unity3D game engine [20]. It can generate close to realistic visual data of the target model under multiple configurable lighting conditions and with or without Earth background. The virtual target was programmed to describe predefined trajectories while the camera was fixed. The camera intrinsic parameters were fixed to match those of the Zero-G Lab data.

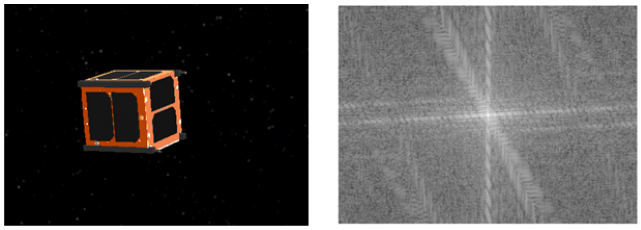
### 3.3 Blender-based Synthetic Data

To increase the diversity of the dataset, we used a second rendering engine for generating data. The Blender-based Synthetic Data was generated using the Blender open-source 3D computer graphics software [1]. The Computer Aided Design (CAD) model of the CubeSat was imported into Blender. The intrinsic camera parameters were adapted to match the physical properties of the camera used in the Zero-G Lab. The pose information from the Zero-G Lab generated datasets were used as groundtruth labels to render the individual images. In Blender, the camera is fixed in origin, and the target is moved relative to the camera based on the pose information. This enables easier image acquisition and verification of the groundtruth labels after rendering.

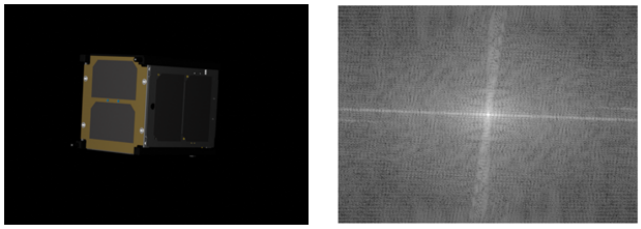
### 3.4 Discussion

The proposed CubeSat-CDT dataset will foster new research on leveraging the temporal information while dealing with the symmetries of a 1U CubeSat, therefore addressing the challenges of estimating the 6-DoF poses of this common platform used in a wide range of new space missions. As presented in Fig. 3, there is a difference in the high-frequency components of the spectrum of the real and synthetic images. Therefore, it is crucial to take into consideration the data domain used for training a DL-based solution. Indeed, the domain gap can lead to a considerable generalization error if it is not addressed.

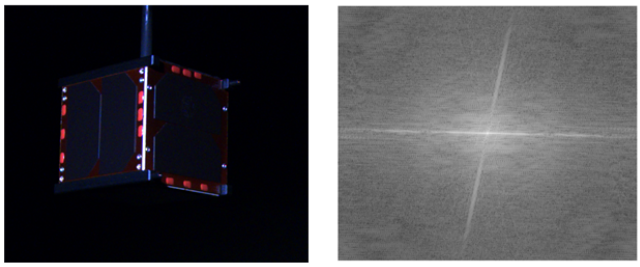
The trajectories defined for the CubeSat were designed to emulate close-range operations when a 1U CubeSat is deployed in orbit. Fig. 4 presents the position and orientation of the trajectories performed in the three datasets. As summarized in Table 2, the 1U CubeSat relative distance to the camera frame varies from 0.40m to 3.8m depending on the dataset. Different illumination conditions were taken into account to emulate solar flares and reflections in the solar panels to better generalize the satellite detection and subsequent pose estimation.



(a) Synthetic (SPARK)



(b) Synthetic (Blender)



(c) Zero-G Lab

Fig. 3: Qualitative comparison of example images in the spatial (left) and frequency (right) domains for (a) synthetic data generated with SPARK, (b) synthetic data generated with Blender, (c) real data acquired in the Zero-G Laboratory.

## 4 Proposed baseline for spacecraft trajectory estimation

In this section, we introduce a baseline to evaluate the effects of both the domain gap and temporal information on the CubeSat trajectory estimation.

### 4.1 Problem formulation

The problem of trajectory estimation consists in estimating the 3D positions and attitudes – *i.e.* orientations – of a target spacecraft in the camera reference frame along the recorded sequence.

Following the notations introduced in [19], let $V_I = \{I_1, \cdots, I_N\}$ be a sequence of RGB images featuring the observed spacecraft, $N$ being the total

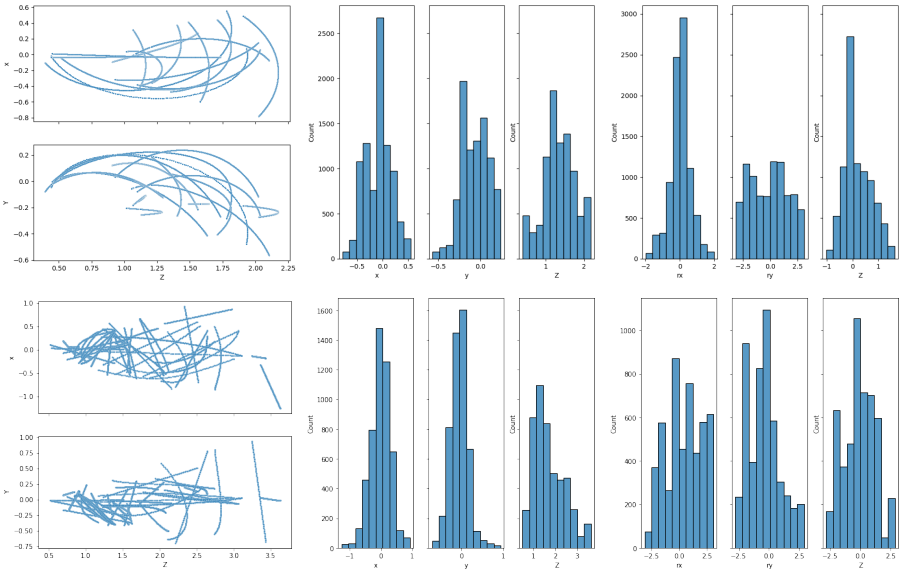


Fig. 4: Trajectory analysis for the Blender & Zero-G Lab (First row) and SPARK (Second row) subsets. From left to right: trajectory analysis in x and y vs z axis, range distribution for all the axes, angle distribution for all the axes.

number of frames. Acquisition is made using a camera with known intrinsics $K \in \mathbb{R}^{3\times3}$. The goal is to estimate the trajectory $\mathcal{Y} = \{(t_1, \mathsf{R}_1), \cdots, (t_N, \mathsf{R}_N)\}$, where $(t_i, \mathsf{R}_i) \in \mathbb{R}^3 \times \mathrm{SO}_3(\mathbb{R})$ is the 3D location $t$ and rotation $R$ of the spacecraft captured at frame $i$.

### 4.2 Proposed Approach

The proposed model is composed of an EfficientNet B2 [29] backbone that takes a sequence of images and processes each frame in parallel then passes the learned features to a TCN model to compute the 6-DoF poses over the full sequence.

**Spatial Feature Extraction** For a given video sequence $V_I$, the EfficientNet B2 [29] feature extractor,

$$f : I \in \mathbb{R}^{M\times N} \mapsto Z \in \mathbb{R}^{\Psi}, \tag{1}$$

is applied frame by frame on $V_I$ resulting in a sequence of estimated learned features $\hat{\mathcal{X}} = \{f(I_1), \cdots, f(I_N)\}$. In Equation (1), $\Psi = 128$ is the dimension of the extracted CNN features, $M$ is the image dimension, and $N$ is the number of frames.

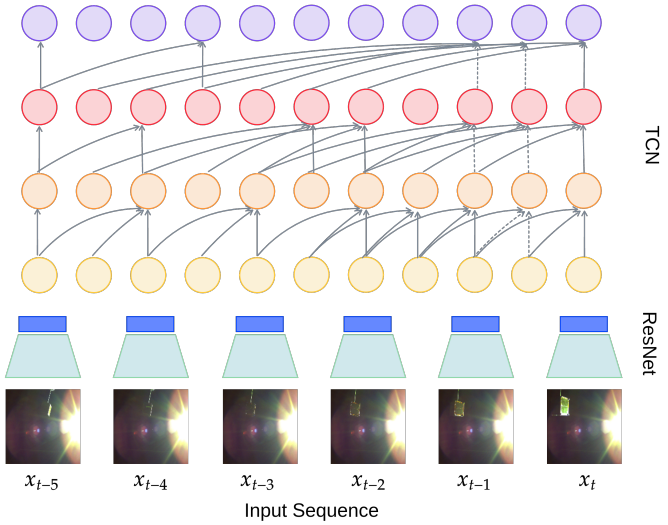



Fig. 5: Our TCN-based trajectory estimation model. At time $t$, we leverage the spatial features of each frame to estimate the target poses using a TCN, leading to smooth and accurate trajectory estimation.

**3D Trajectory estimation** Given a sequence of spatial features $\mathcal{X} \in \mathbb{R}^{\Psi \times N}$, the goal is to lift into the 6D space of poses. To that end, we need to estimate a function $g(\cdot)$ that maps $\mathcal{X}$ to its corresponding 6D sequence, such that:

$$g : \mathcal{X} \in \mathbb{R}^{\Psi \times N} \mapsto \mathcal{Y} \in \left(\mathbb{R}^3 \times \mathrm{SO}_3(\mathbb{R})\right)^N. \tag{2}$$

Independently estimating the poses from each frame would lead to temporally inconsistent results. Therefore, the function $g(\cdot)$ is approximated by a Sequence-to-Sequence Temporal Convolutional Network (*Seq2Seq TCN*) model using 1D temporal convolution. Such a model is illustrated in Fig. 5.

Finally, the sequence of poses is obtained using the composition of the functions $f(\cdot)$ and $g(\cdot)$, such that

$$\hat{\mathcal{Y}} = g \circ f(V_I). \tag{3}$$

### 4.3 Justification of the Proposed Approach

We note that TCNs represent a variation of convolutional neural network for sequence modelling tasks. Compared to traditional Recurrent Neural Networks (RNNs), TCNs offer more direct high-bandwidth access to past and future information. This allows TCN to be more efficient to model the temporal information of the input data with fixed size [18]. TCN can be causal; meaning that there is no information "leakage" from future to past, or non-causal where past and future information is considered. The main critical component of the TCN is the

| Train set \ Test set | Lab (Zero-G) | Synthetic (SPARK) | Synthetic (Blender) |
|---|---|---|---|
| Lab (Zero-G) | 0.05m / 12.98° | 0.40m / 115.27° | 0.50m / 86.77° |
| Synthetic (SPARK) | 0.26m / 92.25° | 0.15m / 72.38° | 0.60m / 126.31° |
| Synthetic (Blender) | 0.30m / 102.14° | 0.47m / 127.59° | 0.14m / 88.30° |

Table 3: Pose MSE when regressed frame per frame independently, for the three data domains.

| Data domain \ Model | Temporal | Single frame |
|---|---|---|
| Lab (Zero-G) | 0.02m / 11.27° | 0.05m / 12.98° |
| Synthetic (SPARK) | 0.10m / 105.7° | 0.15m / 72.38° |
| Synthetic (Blender) | 0.08m / 54.27° | 0.14m / 88.30° |

Table 4: Pose MSE when regressed by the Temporal Convolutional Network for the three data domains.

dilated convolution layer [11], which allows to properly treat temporal order and handle long-term dependencies without an explosion in model complexity. For simple convolution, the size of the receptive field of each unit - block of input which can influence its activation - can only grow linearly with the number of layers. In the dilated convolution, the dilation factor $d$ increases exponentially at each layer. Therefore, even though the number of parameters grows only linearly with the number of layers, the effective receptive field of units grows exponentially with the layer depth.

Convolutional models enable parallelization over both the batch and time dimension while RNNs cannot be parallelized over time [2]. Moreover, the path of the gradient between output and input has a fixed length regardless of the sequence length, which mitigates the vanishing and exploding gradients. This has a direct impact on the performance of RNNs [2].

## 5 Experiments

To analyse further the features of the proposed CubeSat-CDT dataset, we conducted two sets of experiments. First, we analyse the gaps between the three different domains by focusing only on single-frame CubeSat pose estimation. Second, we demonstrate the importance of leveraging the temporal information for more accurate predictions.

As presented in Table 4, our proposed method reduces the pose prediction error, on average by a factor of 2. Furthermore, we note that our approach provides a smoother and a more temporally coherent trajectory as highlighted in Fig. 6 and 7.

We use a PoseNet model [14] with an EfficientNet [29] backbone for feature extraction, followed by fully connected layers for pose regression.

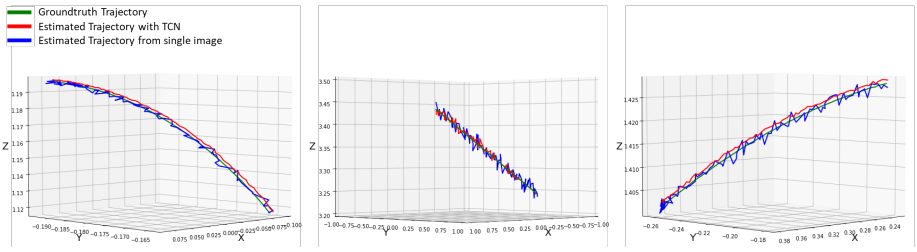



Fig. 6: Groundtruth trajectories of the CubeSat center (in green), estimated trajectories using TCN (in red) and estimated positions using single-frame regressions (in blue). From left to right: example sequences from Zero-G Lab, SPARK and Blender.

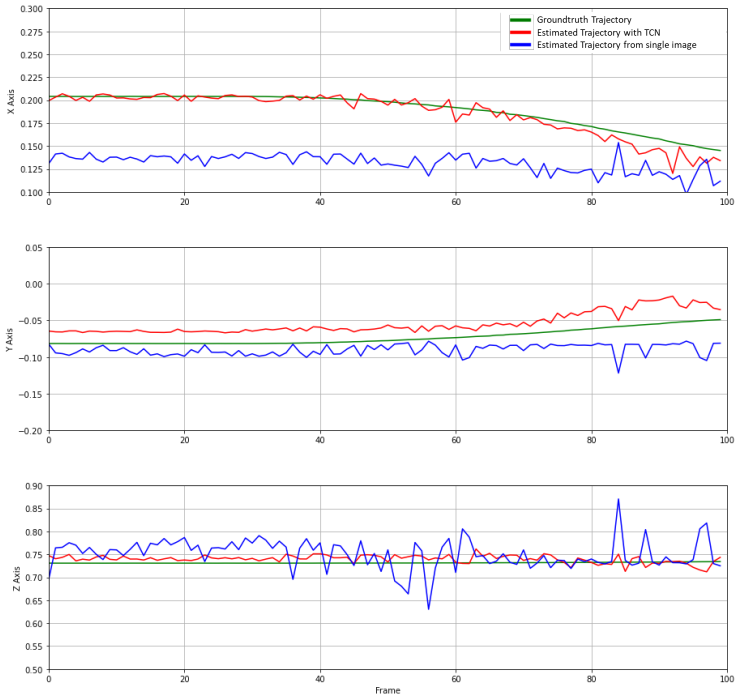



Fig. 7: Per Axis position estimations for *Sequence*7 from Zero-G Laboratory data.

### 5.1 Domain Gap Analysis

In the first experiment, we trained our pose estimation model in a cross-validation manner – *i.e.,* training on one domain subset and testing on another one – in order to assess the gaps between the different domains. When training and testing on the same subset, we used a 80%-20% split of the data. The temporal infor-

mation is not used to evaluate the impact of the domain in this setup, so that the pose of the 1U CubeSat is regressed by single frame processing. To eliminate the temporal dependency in the datasets, training data was randomly shuffled.

The quantitative results, presented in Table 3, confirm the relevance of the cross-domain data and the need for applying techniques to minimize the gap between domains. The best inter-domain results were obtained when training the model on the SPARK synthetic data then testing on the Zero-G Lab data (0.26m / 92.25°), with an average position error increment of 0.11m (0.15m / 72.38°when testing on SPARK). Due to the symmetric nature of the satellite under test, the orientation error is considerably high. Indeed, there is not enough information to discriminate between the different CubeSat faces.

### 5.2 Impact of Temporal Information

The second experiment was designed to assess the impact of temporal information on pose estimations. The model proposed in Section 4 was trained on batches of randomly selected images from the training dataset as an input. The images were then split into 80% for training and 20% for testing.

Given the sequence of estimated learned features computed during the first stage, the second part of the model applies a Seq2Seq TCN model using 1D temporal convolution, to produce the 6-DoF poses of the full sequence.

## 6 Conclusions

In this paper, we investigated the problem of trajectory estimation of a symmetric spacecraft using only RGB information. We collected 21 real CubeSats trajectories in a laboratory environment along with two different synthetic datasets generated in Unity (50) and Blender (15). We proposed a model composed of an EfficientNet B2 backbone to process a sequence of frames in parallel and then pass the learned features to a Temporal Convolutional Network to compute the final 6-DoF poses. Our experimental results show the importance of leveraging the temporal information to estimate the pose of an object in space and increase accuracy compared to a direct pose regression per frame. Furthermore, the results demonstrate the relevance of the data domain used to train the proposed model on the final performance. The CubeSat Cross-Domain Trajectory dataset will be publicly shared with the research community in order to enable further research on minimizing the domain gap between synthetic and real data, leveraging temporal information for pose estimation and computing the pose of highly-symmetric objects.

## References

1. Blender 3.0 reference manual — blender manual, `https://docs.blender.org/manual/en/latest/index.html`

2. Bai, S., Kolter, J.Z., Koltun, V.: An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. arXiv:1803.01271 (2018)
3. Biesbroek, R., Aziz, S., Wolahan, A., Cipolla, S.f., Richard-Noca, M., Piguet, L.: The clearspace-1 mission: Esa and clearspace team up to remove debris. In: Proc. 8th Eur. Conf. Sp. Debris. pp. 1–3 (2021)
4. Black, K., Shankar, S., Fonseka, D., Deutsch, J., Dhir, A., Akella, M.R.: Real-time, flight-ready, non-cooperative spacecraft pose estimation using monocular imagery. arXiv preprint arXiv:2101.09553 (2021)
5. Corona, E., Kundu, K., Fidler, S.: Pose estimation for objects with rotational symmetry. IEEE International Conference on Intelligent Robots and Systems pp. 7215–7222 (12 2018). https://doi.org/10.1109/IROS.2018.8594282
6. Dung, H.A., Chen, B., Chin, T.J.: A spacecraft dataset for detection, segmentation and parts recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops. pp. 2012–2019 (June 2021)
7. D'Amico, S., Bodin, P., Delpech, M., Noteborn, R.: Prisma. In: Distributed Space Missions for Earth System Monitoring, pp. 599–637. Springer (2013)
8. Garcia, A., Musallam, M.A., Gaudilliere, V., Ghorbel, E., Al Ismaeil, K., Perez, M., Aouada, D.: Lspnet: A 2d localization-oriented spacecraft pose estimation neural network. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops. pp. 2048–2056 (June 2021)
9. GSFC: NASA's Exploration & In-space Services, `https://nexis.gsfc.nasa.gov/`, publisher: NASA
10. Haas, J.K.: A history of the unity game engine (2014)
11. Holschneider, M., Kronland-Martinet, R., Morlet, J., Tchamitchian, P.: A real-time algorithm for signal analysis with the help of the wavelet transform. In: Wavelets, pp. 286–297. Springer (1990)
12. Hu, J., Ling, H., Parashar, P., Naik, A., Christensen, H.I.: Pose estimation of specular and symmetrical objects. CoRR **abs/2011.00372** (2020)
13. Hu, Y., Speierer, S., Jakob, W., Fua, P., Salzmann, M.: Wide-depth-range 6d object pose estimation in space. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15870–15879 (2021)
14. Kendall, A., Grimes, M., Cipolla, R.: Posenet: A convolutional network for real-time 6-dof camera relocalization. In: Proceedings of the IEEE international conference on computer vision. pp. 2938–2946 (2015)
15. Kisantal, M., Sharma, S., Park, T.H., Izzo, D., Märtens, M., D'Amico, S.: Satellite pose estimation challenge: Dataset, competition design, and results. IEEE Transactions on Aerospace and Electronic Systems **56**(5), 4083–4098 (2020)
16. Lepetit, V., Fua, P.: Monocular model-based 3d tracking of rigid objects: A survey. Found. Trends Comput. Graph. Vis. **1**(1) (2005)
17. Marchand, E., Chaumette, F., Chabot, T., Kanani, K., Pollini, A.: Removedebris vision-based navigation preliminary results. In: IAC 2019-70th International Astronautical Congress. pp. 1–10 (2019)
18. Mishra, N., Rohaninejad, M., Chen, X., Abbeel, P.: A simple neural attentive meta-learner (2017)
19. Musallam, M.A., del Castillo, M.O., Al Ismaeil, K., Perez, M.D., Aouada, D.: Leveraging temporal information for 3d trajectory estimation of space objects. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops. pp. 3816–3822 (October 2021)
20. Musallam, M.A., Gaudilliere, V., Ghorbel, E., Al Ismaeil, K., Perez, M.D., Poucet, M., Aouada, D.: Spacecraft recognition leveraging knowledge of space environment:

Simulator, dataset, competition design and analysis. In: 2021 IEEE International Conference on Image Processing Challenges (ICIPC). pp. 11–15. IEEE (2021)
21. Park, T.H., Märtens, M., Lecuyer, G., Izzo, D., D'Amico, S.: Speed+: Next generation dataset for spacecraft pose estimation across domain gap. arXiv preprint arXiv:2110.03101 (2021)
22. Park, T.H., Sharma, S., D'Amico, S.: Towards robust learning-based pose estimation of noncooperative spacecraft. arXiv preprint arXiv:1909.00392 (2019)
23. Pauly, L., Jamrozik, M.L., Del Castillo, M.O., Borgue, O., Singh, I.P., Makhdoomi, M.R., Christidi-Loumpasefski, O.O., Gaudilliere, V., Martinez, C., Rathinam, A., Hein, A., Mendez, M.O., Aouada, D.: Lessons from a space lab – an image acquisition perspective (2022). https://doi.org/10.48550/ARXIV.2208.08865, https://arxiv.org/abs/2208.08865
24. Pellacani, A., Graziano, M., Fittock, M., Gil, J., Carnelli, I.: Hera vision based gnc and autonomy. 8 TH EUROPEAN CONFERENCE FOR AERONAUTICS AND SP (2019). https://doi.org/10.13009/EUCASS2019-39
25. Pitteri, G., Ramamonjisoa, M., Ilic, S., Lepetit, V.: On object symmetries and 6d pose estimation from images. In: 2019 International Conference on 3D Vision, 3DV 2019, Québec City, QC, Canada, September 16-19, 2019. pp. 614–622. IEEE (2019)
26. Proença, P.F., Gao, Y.: Deep learning for spacecraft pose estimation from photorealistic rendering. In: 2020 IEEE International Conference on Robotics and Automation (ICRA). pp. 6007–6013. IEEE (2020)
27. Robots, U.: Ur10, https://www.universal-robots.com/products/ur10-robot/, accessed on 2022-03-11
28. Taketomi, T., Uchiyama, H., Ikeda, S.: Visual SLAM algorithms: a survey from 2010 to 2016. IPSJ Trans. Comput. Vis. Appl. **9**, 16 (2017). https://doi.org/10.1186/s41074-017-0027-2
29. Tan, M., Le, Q.V.: Efficientnet: Rethinking model scaling for convolutional neural networks. In: Chaudhuri, K., Salakhutdinov, R. (eds.) Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA. Proceedings of Machine Learning Research, vol. 97, pp. 6105–6114. PMLR (2019)
30. University, S.: Space rendezvous laboratory, https://damicos.people.stanford.edu/, accessed on 2022-03-11