UNIVERSITÉ DU LUXEMBOURG

PhD-FSTM-2024-027
The Faculty of Science, Technology and Medicine

DISSERTATION

Defence held on 12/04/2024 in Luxembourg

to obtain the degree of

DOCTEUR DE L'UNIVERSITÉ DU LUXEMBOURG

by

Mohamed Adel Mohamed ALI
Born on August 26, 1993 in Red Sea Governorate – Egypt

ADVANCING VISION-BASED SPACE SITUATIONAL AWARENESS: FROM DATA TO REPRESENTATION LEARNING

Dissertation defence committee

Dr. Djamila Aouada, dissertation supervisor
Assistant Professor, Université du Luxembourg

Dr. Bhavani Shankar Mysore, Chairman
Assistant Professor, Université du Luxembourg

Dr. Holger Voos.
Professor, Université du Luxembourg

Dr. Torsten Sattler
Senior Researcher, Czech Technical University in Prague

Dr. Zuzana Kúkelová
Assistant Professor, Czech Technical University in Prague

Affidavit

I hereby confirm that the PhD thesis entitled "Advancing Vision-Based Space Situational Awareness: From Data to Representation Learning" has been written independently and without any other sources than cited.

Luxembourg, 28/05/2024 Mohamed Adel Mohmaed ALI

Name

Acknowledgements

First and foremost, I would like to thank my doctoral advisor Dr. Djamila Aouada for her exceptional support and supervision throughout my doctoral studies. Her kindness, scientific curiosity, and enthusiasm were pivotal to my work's success. I am immensely thankful for her continuous advice and the autonomy she provided, allowing me to explore my research interests freely.

Our initial encounter occurred during a lecture Dr. Aouada presented at ViBot, the University of Burgundy. Before knowing about her research group or the University of Luxembourg, I was certain that she was the mentor I desired to guide my academic and professional paths. I consider myself fortunate to have met individuals in my life who have illuminated my path, goals, and the means to achieve them. Working with Dr.Aouada has been one of these fortuitous encounters. She has not only welcomed me as a student but also played a crucial role in helping me discover my true passions. For this, I am deeply grateful.

I would like to express my deepest gratitude to Dr. Vincent Gaudillière who through his patience, meticulous attention to detail, and warm personality, has significantly contributed to my professional growth and development. Indeed his guidance was invaluable in my collaborative endeavors, offering countless insightful discussions that grounded and directed my roaming search for research ideas.

His constant openness, keeping his office door always open for me, always willing to discuss any research challenges I faced, often provided me with a sense of direction. His approachability and willingness to share his knowledge have been a cornerstone of my academic journey.

I am grateful to Dr. Kassem Al Ismaeil for his vital role at the beginning of my PhD

journey. Holding my feet to the fire, pushing me to focus more on my work. His influence was essential in forming a strong foundation for both my academic endeavors and personal growth.

I extend my heartfelt thanks to Dr. Renato Baptista and Dr. Enjie Ghorbel for their mentorship and support during my first internship, which marked my first step into academic research. Despite my lack of experience, they were instrumental in guiding me through the difficulties of the field, laying a solid foundation for my future endeavors and helping me find my path in academic research.

I am deeply grateful of the opportunity to be a member of the CVI^2 lab, which I proudly consider my team. My sincere gratitude goes to all those who have contributed to establishing and fostering this exceptional research environment. Their efforts have profoundly influenced both my academic and personal journey, for which I am truly thankful.

And for Chrysanthi Katrini as the butterfly effect cannot be seen, the butterfly effect will not fade. And you, your deeds are something I will always carry a deep gratitude for.

I wish to express my deepest gratitude to my family for their unconditional love, invaluable advice, and for believing in me even when I doubted myself. The scientific and engineering discussions with my father Adel since my childhood, my mother Manal's unwavering belief in me at times when I lacked faith in myself, and the shared curiosity with my brothers Asem and Hazem have all profoundly shaped my way of thinking. Their collective support and belief have been a cornerstone of my journey, leaving an indelible mark on my personal and academic development.

Finally, my sincere appreciation to our industrial partner "LMO" for their support and insights, which significantly enhanced my research and bridged theoretical concepts with practical applications. This partnership not only advanced my academic journey but also established a solid foundation for my future endeavors. I'm truly thankful for their guidance and the opportunities this collaboration has offered.

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).

Sight recognizes visible objects and it perceives many of them and of the visible properties by recognition. Thus it recognizes a man to be a man, and a horse to be a horse and Zayd himself to be Zayd, if it has seen them previously and remembers having seen them.

Recognition may be of an individual object or of a species. Recognition of an individual object occurs as a result of likening the form of the visible object which the sight perceives at the time of recognition to the form it has formerly perceived of it. Recognition of a species occurs as a result of likening the form of the visible object to that of similar individuals of the same species which the sight has formerly perceived.

When sight perceives a form of which it has previously had perception, or of forms like it, it will immediately perceive what the form is in consequence of its perception of some of the features in that form, if it remembers its former perception of that form or of those like it.

So, that which is perceived by recognition is perceived by signs, but not everything perceived by inference is perceived by signs.

– **Abu Ali al-Hasan ibn al-Haytham (965–1040 AD)**

The Book of Optics: On Direct Vision - Book II. On The Visible Properties, Their Causes And The Manner of Their Perception

Index

List of Figures

List of Tables

Nomenclature

Abbreviations

ADR Active Debris Removal

CNN Convolution Neural Network

CV Computer Vision

DL Deep Learning

EO Electro-Optical

ESA European Space Agency

$G-CNNs$ Group Convolution Neural Network

GNC Guidance, Navigation, and Control

$LiDAR$ Light Detection and Ranging

OOS On-Orbit Servicing

PnP Perspective-n-Point

$RANSAC$ Random Sample Consensus

SSA Space Situational Awareness

TOF Time-Of-Flight

VBN Vision Based Navigation

Notations

K Intrinsic Camera Matrix

P Camera Pose Matrix in 3D

R Rotation Matrix in 3D

t Translation Vector in 3D

E(n) Euclidean Group

SE(n) Special Euclidean group

SO(n) Special Orthogonal Group

p Point in 2D image

X Point in 3D

Summary

The expansion of space activities, including the deployment of CubeSats and low-cost satellites, is transforming our interaction with space and enhancing applications such as remote sensing, navigation, and telecommunication. However, this growth has led to an increase in satellites and debris, raising collision risks in Earth's crowded orbits and questioning the sustainability of space utilization. In response, space agencies have focused on mitigating these risks through Active Debris Removal (ADR) and extending satellite lifespans via On-Orbit Servicing (OOS). A key challenge in these initiatives is the safe approach and capture of non-cooperative debris, requiring precise determination of their position and orientation (pose) relative to servicing spacecraft. Recent missions have begun integrating passive monocular cameras to improve the accuracy and reliability of navigation systems, directly informing the focus of this thesis.

This thesis explores the application of machine learning in enhancing visual tasks critical for Space Situational Awareness (SSA), focusing on the development and evaluation of monocular vision-based pose estimation systems. It specifically focuses on the use of direct end-to-end learning approaches over hybrid modular approaches, highlighting their potential to enhance SSA operations.

Addressing the challenge of improving accuracy in direct pose estimation, we introduce the use of Group Equivariant Convolutional Neural Networks (G-CNNs). Our contribution leverages G-CNNs to enable a more effective capture of spatial and geometric information critical for estimating relative and absolute pose estimation . Our method surpasses traditional CNNs in handling the complex geometries of space objects. This enhances space situational

awareness with more accurate pose estimation, crucial for space safety and sustainability, while also reducing model sizes for greater computational efficiency.

To address the unique challenges of the space environment, this thesis develops vision models that maintain robustness against appearance variations while being sensitive to geometric transformations. Through self-supervised learning, it constructs image representation that are invariant to appearance changes while being sensitive to geometric transformations. Independent of human-annotated data, this methodology is effective for visual place recognition across diverse conditions, underscoring its potential for broad application beyond space navigation.

Additionally, a novel method for estimating 3D trajectories of space objects from single RGB camera footage is introduced. This approach is vital for Space SSA, ADR, and OOS missions. It enhances precision and stability in trajectory estimates by integrating temporal data. This improvement is an important step forward in spatial tracking for SSA missions.

Finally, we contribute to the field by curating a specialized dataset aimed at training deep learning-based computer vision models specifically designed to tackle SSA challenges.

Highlights our role in initiating the SPARK competition, aimed at advancing research and innovation in space target recognition and detection. This competition not only guided the development of several datasets but also offered a practical application framework for evaluating the performance of models trained on this dataset.

By interweaving these studies, this thesis presents a narrative that encapsulates the pressing need for advanced machine-learning solutions in the face of the intricate challenges posed by space exploration and monitoring.

Chapter 1

Introduction and Motivation

The frequency of satellite launches has significantly increased in recent years. Driven by a diverse range of missions, each satellite is characterized by specific size, functionalities, and lifespan [1, 2]. While this increase in satellite deployment advances global communications and research, it also presents significant challenges.

Despite being meticulously designed for their intended mission duration, satellites are vulnerable to unexpected anomalies and malfunctions. These unforeseen issues can abruptly convert active satellites into hazards, threatening the integrity of the existing orbital ecosystem. Moreover, the hostile space environment, characterized by extreme temperatures and illumination conditions, radiation, and high-velocity debris, further worsen these challenges, turning operational difficulties into critical concerns that demand prompt and effective solutions.

In this context, there is an emerging and critical demand for specialized orbital missions, particularly focused on On-Orbit Servicing (OOS) [4] and Active Debris Removal (ADR) [5]. These missions are pivotal for maintaining space safety and sustainability. OOS missions encompass a range of activities, including in-space inspection, repair, and satellite maintenance. They offer commercial applications ranging from extending satellite lifespans to assisting in extravehicular activities [4].

Simultaneously, ADR missions are strategically implemented to mitigate space debris by removing non-functional objects from orbit, as depicted in Figure 1.1. Such interventions are

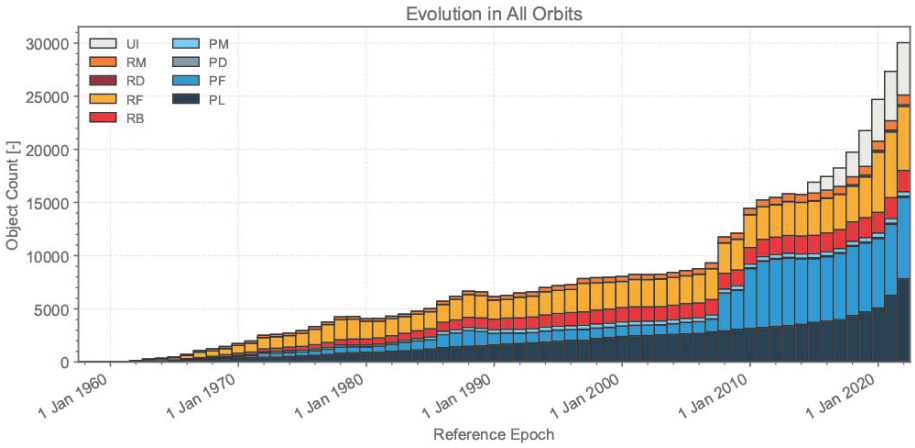


Figure 1.1: Image from ESA's Space Environment Report 2022 [3]. [PL = Payload (usually one or many satellites that a rocket launches to space); PF = Payload Fragmentation Debris; PD = Payload Debris; PM = Payload Mission Related Object; RB = Rocket Body; RF = Rocket Fragmentation Debris; RD = Rocket Debris; RM = Rocket Mission Related Object; UI = Unidentified.]

critical for lowering the probability of in-orbit collisions, which in turn curtails the potential for severe and potentially exponential escalations in space debris, as described in [5].

The advancement and successful implementation of such missions are not just theoretical concepts. Several successful technology demonstrations have already been executed, like the PROBA-3 [6] by the European Space Agency (ESA) and commercial missions like MEV-1 [7] by Northrop Grumman. Furthermore, future endeavours like Clearspace-1 [8] are currently in development, poised to further contribute to this rapidly evolving field of space technology. These missions are primarily focused on the disposal or maintenance of defunct or non-operational space entities, known as *target*, using operational spacecraft, termed *chaser*.

A fundamental task in these missions is the accurate determination of the target's pose - *i.e.* position and orientation - relative to the chaser. This challenge is intensified in ADR/OOS missions where the target can be uncooperative, which may lack navigational aids such as visual markers or LEDs and are sometimes completely non-functional, as referenced in [9]. In contrast, missions involving cooperative objects typically feature targets equipped with such navigational aids, facilitating the tracking and approach process.

For pose estimation, several sensors may be available, including Monocular RGB/Greyscale Cameras, Stereo Cameras, Thermal Cameras, RADAR, and LiDAR. Monocular cameras are of particular interest due to their compact size, low power consumption, and ease of integration. However, most mission plans and technology demonstrations tend to combine single and stereo cameras with LiDAR to address the limitations of using a monocular camera exclusively. Despite these challenges, the proven effectiveness of monocular systems in terrestrial applications has driven significant research into their use for cost-effective ADR/OOS missions [10].

Missions like AVANTI [11] and RemoveDEBRIS [12] have provided valuable flight results for autonomous navigation systems relying on monocular cameras, underscoring the necessity of enhancing the resilience of these systems for the success of ADR and OOS missions. Thus, there is an urgent need for innovation in autonomous navigation technologies to improve safety, precision, and reliability.

1.1 On-Orbit Servicing Technology

The concept of OOS is rooted in the space race's early days. The initiative began in 1959 when the United States Air Force launched the Space Logistics, Operations, Maintenance, and Rescue (SLOMAR) [13] an illustration of the proposed spacecraft is shown in Figure (1.2). This pioneering investigations aimed at developing designs for human-crewed spacecraft that could support military space stations. As part of the broader 'Space Development Planning Study' by the USAF, encompassing ten diverse studies, SLOMAR was a significant step towards advanced space capabilities. They covered various topics, from satellite interception and global surveillance to developing recoverable space launch vehicles and lunar missions. OOS missions, integral to space operations today, primarily focus on disposing or servicing defunct or non-operational space entities (referred to as *targets*) using operational service spacecraft referred to as *chaser*. With the escalating value and intricacy of orbital assets and the pressing challenge of orbital debris, autonomous OOS capabilities are becoming increasingly imperative. Autonomous systems are essential for enhancing operational

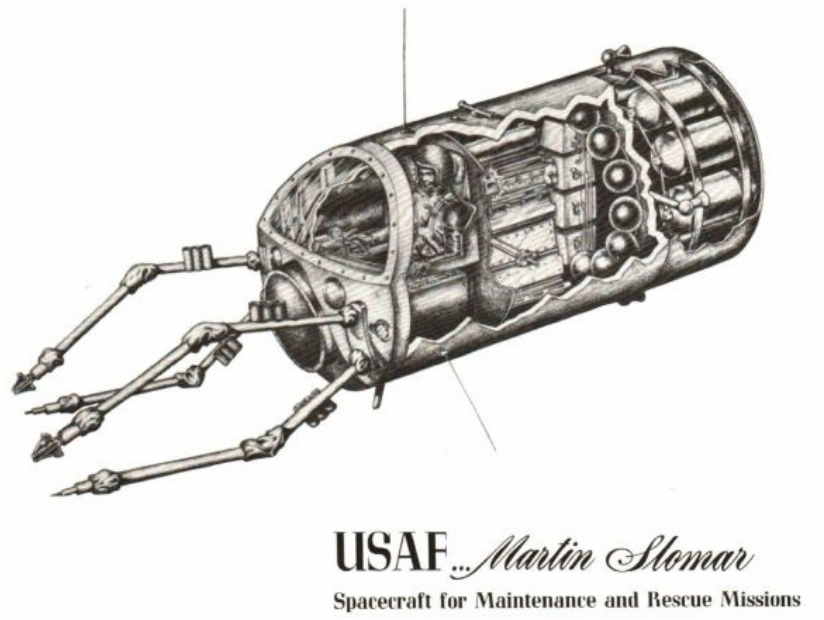


Figure 1.2: A one-man SPACE TUG is one of the vehicles being studied by The Martin Company SLOMAR team [13]. A large space station would be assembled in orbit by a man in the tug using remotely controlled power tools and manipulators.

safety, efficiency, and the sustainability of the space environment. They facilitate intricate servicing missions without the extensive risks and costs associated with human spaceflight. The foundational ideas conceptualized in programs like SLOMAR have evolved, paving the way for the current need for autonomous OOS operations that can independently manage the complexities of modern space servicing tasks. These would enable more efficient, safe, and sustainable space operations, as they navigate the intricate processes involved in servicing missions. The structure of these missions, marks a progression from the foundational concepts of SLOMAR to the sophisticated, autonomous systems needed today, identified by Colmenarejo *et al.* in [14]:

- **Phasing:** During this initial stage, the chaser spacecraft calculates the target's orbital details and adjusts its own orbital path to match that of the target.
- **Approach:** This intermediate stage sees the chaser drawing closer to the target from distances of several kilometers to a few meters, undertaking various proximity ma-

neuvers like orbiting around the target, performing inspections, and executing precise approach techniques. Should the target be spinning, this stage culminates with the chaser either stabilizing the target's rotation or synchronizing its movement to match the target's rotational axis.

- **Capture:** The final stage involves the physical securing of the target by the chaser, which can be through a solid or flexible link. The subsequent action, whether it is removing the target from orbit (ADR) or providing maintenance (OOS), depends on the primary mission goal.

While the phasing stage involves direct maneuvers to align orbits and may include ground-controlled operations, the subsequent approach and capture/refueling stages are predominantly governed by the relative movements of the chaser and the target. These stages typically require advanced autonomous guidance, navigation, and control (GNC) capabilities, highlighting the complexity and precision required in modern space missions.

1.2 Vision-Based Spacecraft Pose Estimation

The integration of autonomous systems in ADR and OOS missions is increasingly essential due to the complex nature of space operations. During these missions, the role of the chaser spacecraft becomes critical, especially when engaging with space objects that require disposal or maintenance. The approach phase, where the chaser aligns itself with a rapidly moving target for the capture phase, presents a significant challenge due to the need for precise and safe maneuvering.

Challenges arise from delays in ground-controlled operations and the limited availability of ground stations, which can impede tasks required for close-range operations. Consequently, there is an escalating demand for autonomous navigation systems that enhance current operational standards, ensuring approaches are executed with utmost safety, accuracy, and reliability, all within the limits of available computing resources [15].

Estimating the relative pose of a spacecraft consists of calculating the position and attitude of a target spacecraft in relation to a chaser spacecraft based on direct measurements

Figure 1.3: An illustration of a chaser spacecraft employing tracking and monitoring systems to assess the relative position and orientation of a target spacecraft.

from one or more sensors, as illustrated in Figure (1.4). Optical sensors used for this purpose can be categorized into active and passive types. Active sensors, such as LiDAR and Time-Of-Flight (TOF) cameras, emit their own light source, whereas passive sensors, like monocular and stereo cameras, rely on existing ambient light. In the realm of spacecraft relative navigation, Electro-Optical (EO) sensors, particularly stereo cameras [16, 17] and LiDAR [18], are frequently employed alongside monocular cameras. This combination helps to overcome the limitations in range information inherent to single monocular cameras.

Pose estimation systems using monocular cameras are gaining popularity over those with active sensors or stereo cameras, primarily due to their lower mass, energy needs, and simplicity [19, 20]. Despite this, achieving a robust and precise monocular vision-based navigation system is challenging. Extracting visual features that can be used to estimate geometric configuration of the target is crucial in this process, demanding advanced image processing and computer vision techniques.

Over the past decade, vision-based spacecraft pose estimation has predominantly relied

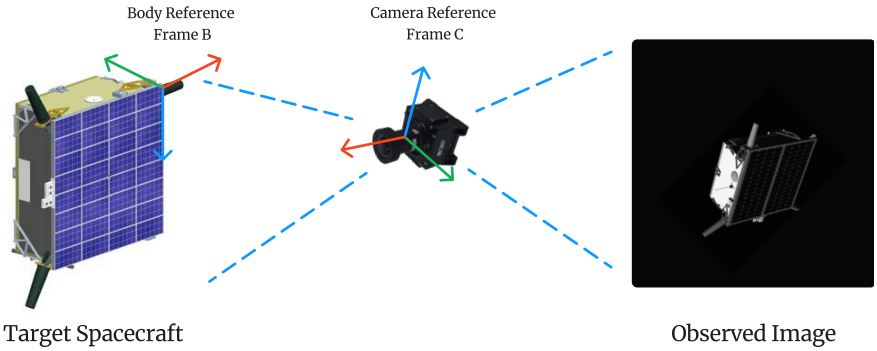


Figure 1.4: **Spacecraft Pose Estimation Illustration** – In the context of spacecraft observed through a camera, let B and C denote the body-fixed frames of the target spacecraft and the observing camera attached to the chaser, respectively. formal statement of this problem is to estimate the relative pose of B in relation to C. This pose is characterized by a relative position vector $\mathbf{t}_{CB}$ and a unit quaternion $\mathbf{q}_{BC}$. Here, $\mathbf{t}_{BC}$ represents the coordinates of the origin of B expressed in the frame C, and $\mathbf{q}_{BC}$ denotes the quaternion that defines the rotation required to align frame B with frame C. This elaboration mathematically formalises the relationship between the target spacecraft and the camera in terms of their respective body-fixed reference frames, employing both quaternion representation for relative orientation and vector notation for relative position.

on the use of hand-engineered features, which are described with feature descriptors and detected via feature detectors. These features are identified in 2D images, and their 3D correspondences are utilized to ascertain the spacecraft's relative pose [21]. Despite the effectiveness of perspective transformations in facilitating the convergence of pose solutions through feature correspondences, these hand-engineered features exhibit a lack of robustness under the extreme lighting conditions found in space [22].

Feature-based methods for spacecraft pose estimation have struggled under the challenging conditions of space imagery, including variable lighting, low signal-to-noise ratios, high contrast, and the intricate structures of the targets whether its symmetries of the spacecraft or low texture [19, 10]. These limitations often lead to inaccurate estimations of the spacecraft's state in various situations.

These extracted features are then matched with a 3D model of the target to estimate its pose. The pose estimation system can be either model-based, utilizing a pre-existing offline

3D model, or model-free, requiring on-board reconstruction of the target's model [23].

Furthermore, it is essential to optimize the balance between computational efficiency and accuracy in model-based and model-free systems for their practical application in space missions. Advancements in this field will improve the reliability and precision of monocular vision-based navigation systems and broaden their applicability to a wider range of ADR/OOS missions.

Developing new algorithms and approaches is critical to address these challenges. This development involves leveraging advancements in Deep Learning (DL) and Computer Vision (CV) to train systems to identify and learn valuable features, enhancing their adaptability to various conditions and target characteristics.

The emergence of DL techniques has significantly redefined the landscape of spacecraft pose estimation. By moving beyond the limitations inherent in feature-based approaches, DL has introduced a new wave of methodologies, characterized by both hybrid modular and direct end-to-end strategies [23] which we will discuss more in the next section. This evolution signifies a pivotal shift towards more resilient solutions adept at navigating the unique and demanding conditions encountered in space imagery. The precise identification of a spacecraft's position and orientation is pivotal for the operational success of chaser missions, encompassing tasks such as inspection, repair, or formation flying. Accurate pose estimation is fundamental for executing close-proximity maneuvers efficiently and safely. Additionally, within the broader context of spacecraft operations, these accurate estimations enable the chaser to make timely and informed decisions regarding its trajectory and adjustments, thus facilitating autonomous engagement with the space environment and ensuring the successful execution of mission objectives.

Hybrid Modular Approaches

Hybrid modular approaches to spacecraft pose estimation integrate multiple DL models with classical computer vision techniques. This method consists of a sequence of distinct stages:

1. Object Detection/Localization: the first stage involves detecting the spacecraft within the image and isolating the region of interest. This is typically achieved through DL-

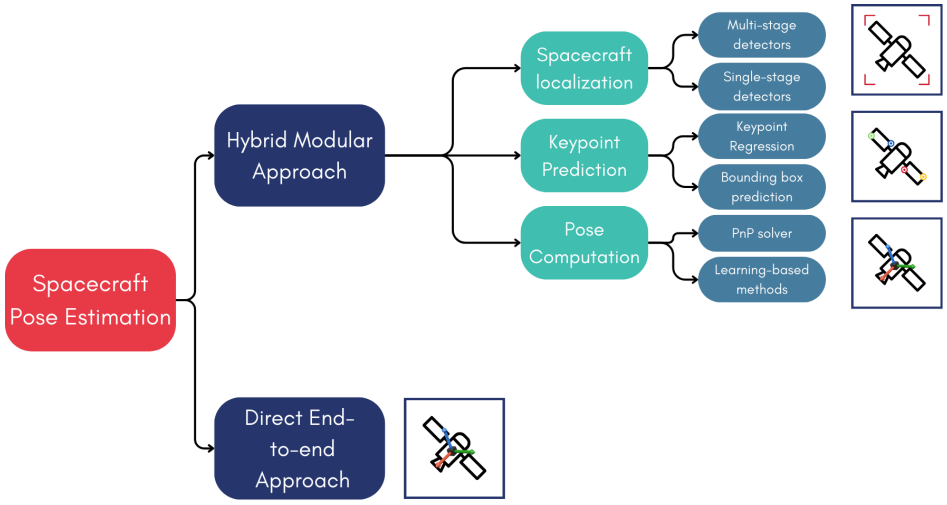


Figure 1.5: Illustration of spacecraft pose estimation methodologies. Upper) Hybrid modular approach, which involve a three-stage process: spacecraft detection/localization, keypoint prediction, and pose computation. The initial stages leverage deep learning, while the final stage incorporates a classical algorithm for outlier elimination crucial to the Perspective-n-Point (PnP) solver and subsequent pose refinement. Lower) Direct end-to-end approach employing single deep learning model to estimate the pose.

based object detection frameworks.

2. Keypoint Prediction: once the spacecraft is localized, the next step is to predict the 2D locations of pre-defined 3D keypoints. DL models are employed to identify these keypoints within the isolated image regions.

3. Pose Computation: the final step is computing the pose using the 2D-3D correspondences of the keypoints identified in the previous stage. Classical algorithms like the Perspective-n-Point (PnP) solver, often coupled with outlier removal techniques like Random sample consensus (RANSAC), are used for the pose estimation.

Direct End-to-End Approaches

Direct end-to-end approaches in monocular pose estimation for spacecraft streamline the estimation process by utilizing a singular DL model to directly derive the spacecraft's pose from images. This technique stands in contrast to hybrid modular approaches by foregoing

the need for multiple processing stages or the incorporation of classical computer vision techniques, thus offering a more direct and efficient pathway to pose estimation.

Key aspects of direct end-to-end approaches include:

1. **Single Model**: Direct end-to-end approaches employ only one DL model to estimate the pose, simplifying the pipeline.

2. Training Methodology: These models are trained using loss functions based on the pose error, focusing solely on the end output.

3. **Intrinsic Learning**: The approach inherently learns the camera parameters during the training process, which means it does not require external inputs like camera parameters or a 3D model of the spacecraft, apart from the ground truth pose labels.

In this thesis, our primary focus centers on exploring and addressing the limitations inherent in direct end-to-end approaches for monocular pose estimation in spacecraft. While these approaches are known for their streamlined process and efficiency, arising from a simplified pipeline, they often exhibit limitations in terms of flexibility and transparency. These limitations can be particularly impactful in complex space mission scenarios. To enhance these direct end-to-end methods, we propose innovative methodologies that leverage various feature learning techniques, aiming to augment their adaptability and analytical clarity.

1.3 Features Learning for Spacecraft Pose Estimation

For spacecraft pose estimation, feature design is critical to capture specific aspects like shape, illumination, and viewpoint variations. Effective feature selection adheres to three core principles, named Hoiem's principles [24]:

1. **Coverage**: Features must encapsulate essential information including color, texture, structural category, position, and surface orientation.

2. **Concision**: Aiming for a minimal yet comprehensive feature set enhances model robustness and generalization, bridging the gap between training and testing performance.

3. **Directness**: Features should be independently predictive for simpler decision boundaries and improved generalization, streamlining the pose estimation process.

These principles underpin an effective model for accurate and reliable spacecraft pose estimation, and will guide our research.

Equivariant and Invariant Features

In this thesis, we focus on the pivotal role of equivariant and invariant features within the domain of deep learning for visual tasks. Equivariant features are designed to adapt to changes in the input, whereas invariant features ensure a consistent output despite such variations. This dynamic interplay between adaptation and consistency is crucial for the successful execution of tasks such as object recognition and pose estimation, forming the core focus of our investigation.

Research Objective I

The first axis of research is that the development of features that exhibit equivariance to a wider range of geometric group transformations, holds significant potential for enhancing the effectiveness of direct pose estimation models specifically designed for space-related applications.

Historical advancements in equivariant / invariant feature development, such as Scale-Invariant Feature Transform (SIFT) [25],Oriented and Steerable filters [26, 27], Rotation-equivariant Fields of Experts (R-FoE) [28], and Lie groups-based filters [29, 30], have significantly contributed to understanding visual patterns through transformations.

Equivariant Features

Despite Convolution Neural Networks (CNNs) inherent translation equivariance, challenges in encoding spatial information persist, especially with local and global pooling. Research indicates that CNN neurons often learn similar features [31, 32]. Moreover, studies have revealed that many neurons within CNNs tend to learn slightly altered versions of similar

fundamental features, such as rotated forms of basic curve, textures, and line detectors, particularly in the initial stages of vision processing [33].

Expanding CNN capabilities involves integrating broader transformation groups. For instance, there have been efforts to make CNNs equivariant to more complex transformations, such as the special Euclidean group, using techniques like scattering transforms with predefined wavelets or B-splines [34, 35, 36, 37]. Similarly, the introduction of group convolutions has allowed networks to achieve equivariance to discrete groups through rotations and flips, enhancing their performance in classification tasks [38].

The utilization of equivariant features in pose estimation is underscored by several essential aspects:

1. **Flexibility in Handling Transformations**: Equivariant models inherently adapt to and consistently recognize variations in an object's orientation or position. This adaptability is essential in geometry inference tasks such as pose estimation, where object orientations can differ widely. The implementation of equivariant features allows models to universally apply their understanding to diverse poses, thereby minimizing explicit learning from each transformed variant. This efficiency not only shortens training duration but also bolsters the precision and robustness of the outcomes [39, 38].

2. **Efficiency in Learning**: Equivariant architectures, particularly CNNs, are highly effective in handling shifts across spatial dimensions, applying learned patterns effectively. This capability is extremely beneficial in pose estimation, where it's essential to understand spatial relationships and orientations. Equivariant networks streamline the learning process by inherently processing various poses of the same object, leading to simpler models with fewer parameters. This results in more compact models with improved performance, which is essential in environments where real-time processing or limited resources are a concern [40, 41].

3. **Reduction of Redundant Data Processing**: Unlike traditional non-equivariant models such as Multilayer Perceptrons, or models with limited equivariance that might need to learn from every possible variation of an input, equivariant models naturally

handle these variations. This feature is particularly useful in complex tasks like pose estimation, which involve a wide range of potential poses. Therefore, equivariant models circumvent the inefficiencies of learning from redundant data [41, 42, 43].

Overall, incorporating equivariant features into models enhances their generalization ability, learning efficiency, and data management, suiting the dynamic nature of computer vision challenges.

Invariant Features

The concept of learning invariant features in computer vision and machine learning is integral to developing models that can effectively adapt to significant changes in appearance. The primary goal in this area is to engineer image representations that capture crucial geometric details of spatial arrangements while remaining unaffected by variations in environmental conditions. This process involves a critical balance between geometric sensitivity and appearance robustness, ensuring that the learned features are both accurate and resilient.

Research Objective II

The second axis of research focuses on the creation of features that demonstrate invariance to a broad spectrum of appearance changes, which is anticipated to substantially improve the performance of computer vision models, particularly those tailored for space-related applications.

The employment of invariant features in the context of various machine learning and computer vision tasks is highlighted by a number of crucial factors:

1. **Stability Against Appearance Variations**: Invariant features in models, especially in CNNs, are designed to maintain consistent recognition despite changes in input appearance. This stability is vital in tasks like object recognition, where the model must reliably identify an object regardless of changes in lighting, angle, or other environmental factors. By focusing on invariant features, models can effectively ”ignore” these

changes, concentrating instead on the underlying, unchanging aspects of the input [44, 45, 46, 47].

2. **Efficiency in Feature Extraction**: Invariant features simplify the feature extraction process by focusing on stable aspects of the input. This approach reduces the complexity of the model, as it doesn't need to learn from every possible variation in appearance. As a result, models with invariant features can be more streamlined and efficient, with fewer parameters to adjust and a quicker training process [48, 49, 47].

3. **Minimization of Overfitting**: By concentrating on invariant features, models are less likely to overfit to specific appearances in the training data. Overfitting occurs when a model becomes too tailored to the training dataset and fails to generalize well to new data. Invariant features help in creating more robust models that can generalize better to unseen data, as they focus on the fundamental, unchanging aspects of the input [50, 51].

In summary, integrating invariant features into machine learning models is essential for tasks requiring stable feature recognition across various appearances. This integration enhances the models' generalization capabilities, efficiency in learning, and reduces the likelihood of overfitting, making them particularly suitable for applications where input data may undergo significant appearance changes.

1.4 3D Trajectory Estimation of Space Objects

The third axis of our research is the integration of temporal information for spacecraft trajectory estimation, as it is essential for estimating 3D trajectories of space objects in aerodynamics and space situational awareness for OOS/ADR missions. This comprehensive approach includes analyzing sequential data cameras and employing neural networks to learn space object dynamics. The enhanced accuracy in trajectory prediction, achieved through nuanced motion pattern understanding and deep learning algorithms, leads to more reliable future position estimations. Temporal coherence in data analysis ensures consistent and smooth

trajectory estimations, while deep learning models adeptly adapt to complex space dynamics. This integration is crucial for robust model development, aiding in critical space operations such as satellite tracking and collision avoidance.

Research Objective III

The third axis of research emphasizes the development of temporal models for spacecraft trajectory estimation. This focus involves creating and refining computational methods that leverage temporal data to predict and analyze the trajectories of spacecraft with enhanced accuracy and reliability, addressing the dynamic and complex challenges posed by space applications.

The advancement of temporal models for spacecraft pose estimation is crucial for several reasons:

1. Improved Prediction Accuracy: The inclusion of temporal data enhances understanding of space object motion patterns, leading to more precise predictions of their positions and trajectories.

2. **Consistency in Trajectory Analysis**: Temporal data contributes to smoother, more consistent trajectory estimates over time, mitigating anomalies or erratic predictions.

3. **Adaptation to Space Dynamics**: The dynamic nature of space requires models that can adapt to changing conditions and behaviors. Temporal data integration equips models to effectively handle these variations.

In summary, the integration of temporal data in the 3D trajectory estimation of space objects is vital for developing robust and accurate models, which is crucial for applications ranging from satellite tracking to collision avoidance in space operations.

Overall, the most critical element in the realm of monocular-based pose estimation systems, particularly for spacecraft, is the pressing requirement to establish an effective feature learning framework. This framework is vital for the accurate estimation of spacecraft poses.

In Figure (1.6) we illustrate the different building blocks for Vision-Based On-Orbit Servicing Stack, with our contribution to each part of the stack, which can serve as a guiding framework for future research in the field.

The initial challenge in this domain is the scarcity of available data for spacecraft pose estimation, necessitating the reliance on synthetically-generated data and advanced simulations to develop effective models.

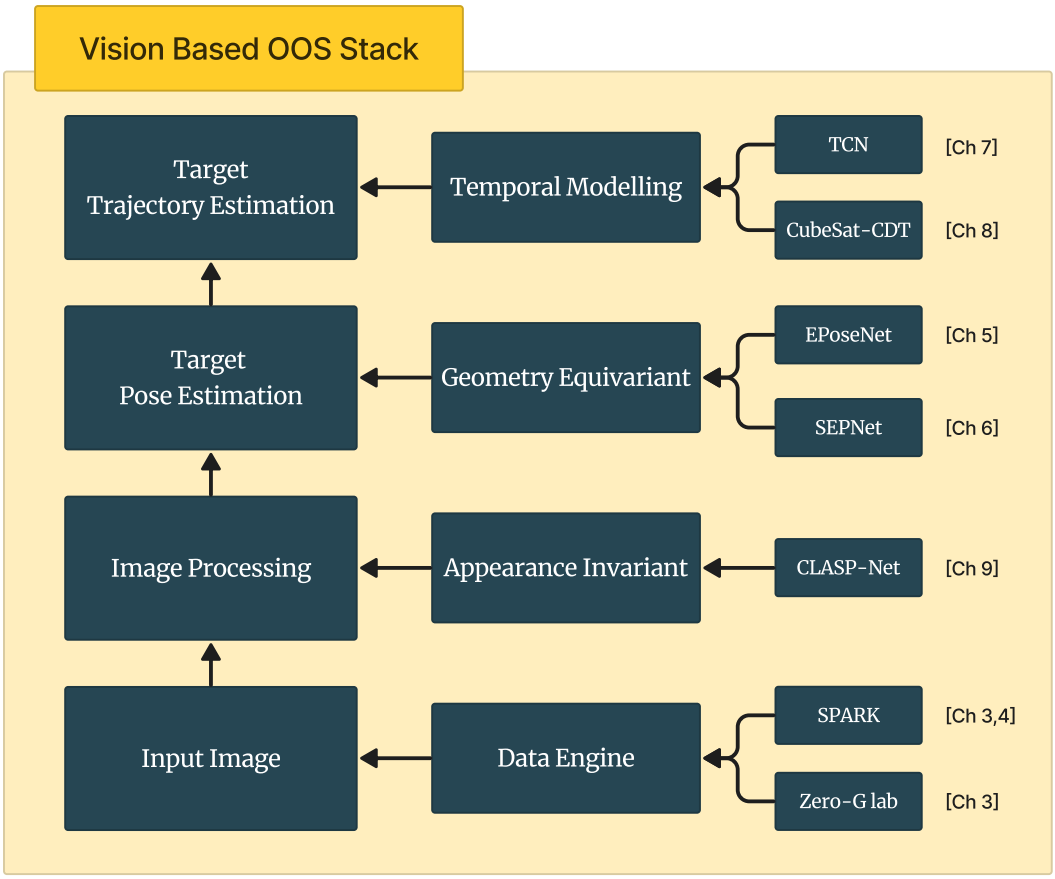


Figure 1.6: **Architectural Diagram of a Vision-Based On-Orbit Servicing (OOS) Stack**: This illustration presents a structured workflow from input image acquisition to target trajectory estimation, highlighting the individual research focus on each module. It encompasses specialized modules for temporal modeling, geometry equivariant feature extraction, and appearance invariance, illustrating our comprehensive, module-specific research within the overarching framework for accurate spacecraft pose estimation.

1.5 Contributions and Thesis Outline

This thesis provides a comprehensive exploration of advanced methods in spacecraft pose estimation and visual place recognition, with a focus on leveraging geometric and appearance features for enhanced space navigation systems.

Chapter (2) establishes the essential groundwork for the research by presenting a comprehensive overview of the background concepts and theories pertinent to the study. This chapter serves as a foundational guide, offering readers a detailed insight into the key principles, methodologies, and theoretical frameworks that underpin the research. It is designed to equip readers with the necessary understanding and context to fully grasp the subsequent chapters, ensuring a cohesive and well-informed exploration of the study's main topics and objectives.

Chapter (3) present our efforts in developing a simulation environment and the integration of both synthetic and real data for a variety of applications. This chapter highlights the methodologies and processes involved in creating a realistic simulation framework. The SPARK simulator, in particular, is instrumental in generating a diverse and extensive dataset of multi-modal images, which plays a crucial role in the development and testing of innovative algorithms. Furthermore we give an overview of the use of the Zero-G lab's contribution to real data acquisition to further complements the synthetic data, offering a benchmark to test our algorithms in more realistic setup.

Chapter (4) provides an in-depth examination of our work on the first SPAcecraft Recognition leveraging Knowledge of space environment (SPARK) challenge. In the 2021 IEEE International Conference on Image Processing (ICIP 2021), the SPARK challenge represented a significant endeavor to promote research and innovation in space target recognition and detection. It details the comprehensive process behind the competition's conception, including the development of the simulator, the creation of a specialized dataset, the intricacies of competition design, and our thorough analysis of the outcomes and insights from the challenge.

Chapter (5) explores the intricate task of end-to-end absolute camera estimation, essential for determining the spatial relationship between a target and chaser spacecraft. The chapter

focuses on a theoretical examination of the use of Equivariant Features for absolute camera pose regression. Absolute pose regression consists in determining the position and orientation of a camera with respect to a 3D world coordinate frame, emphasising practical applications. The goal is to enhance feature learning for direct pose regression techniques by incorporating equivariant features with respect to specific transformation groups. This includes focusing on group equivariant CNN learning methods to develop features that maintain their properties under camera transformations.

Chapter (6) presents the implementation of the novel Equivariant-Pose Net, in the context of spacecraft pose estimation. This application is critical for Space Situational Awareness and on-orbit servicing, where accurate pose estimation is vital. The chapter introduces the Spacecraft Equivariant PoseNet (SEPNet), a deep learning architecture developed specifically for spacecraft pose estimation. SEPNet aims to overcome the limitations of traditional end-to-end methods and conventional CNNs, which often face difficulties in accurately addressing the geometric complexities of pose estimation tasks in orbital missions.

Chapter (7) focuses on the limitations inherent in end-to-end pose estimation approaches, particularly when compared to methods based on 3D geometry. It proposes a novel method employing a translation and rotation equivariant Convolutional Neural Network. This approach explicitly integrates camera motions into the feature space and has shown enhanced performance in standard datasets, highlighting the critical role of geometric information in feature learning.

The focus of Chapter (8) is on the estimation of 3D trajectories of space objects from single-camera RGB video feeds, a key aspect of Space Situational Awareness presented. It details a novel two-stage process that initially identifies the 2D locations of space objects via a convolutional neural network and subsequently extrapolates these into 3D trajectories. This method ensures temporal coherence through a temporal convolutional neural network.

Chapter (9) emphasizing the potential of self-supervised learning in creating robust features that are invariant to varying appearances, particularly suited for the challenges posed by the space environment. This chapter aims to advance features learning capabilities that remain resilient under diverse conditions and are not reliant on human-annotated labels. It

achieves this by integrating contrastive and predictive learning methods, focusing specifically on self-supervised learning techniques to learn features that can be invariant to the unique and dynamic visual aspects of the space environment. This approach can be used for enhancing the adaptability and robustness of navigation and recognition systems in the varying and often unpredictable conditions encountered in space.

Finally in Chapter (10), the thesis reaches its conclusion, where we summarize our main contributions to the field and discuss the open questions that have emerged from our investigation. We also contemplate the broader implications of our research in the context of space situational awareness and spacecraft recognition. Additionally, the chapter identifies potential avenues for future research, underscoring the unresolved challenges and opportunities that our work has brought to light. This final chapter aims to not only encapsulate our achievements but also to inspire continued exploration and innovation in this dynamic and ever-evolving field.

Overall, this thesis weaves together these distinct areas. Focusing on developing equivariant geometric features for direct pose estimation and invariant appearance features for robust visual place recognition. It highlights the importance of temporal coherence in feature learning and combines real and synthetic data to improve the accuracy and reliability of navigation systems in the challenging space environment.

1.6 Publications

The work presented in this Thesis has been disseminated in the proceedings of several international peer-reviewed conference and workshops. A journal submission is also in preparation.

1. **Mohamed Adel Musallam**, Vincent Gaudillière, Enjie Ghorbel, Kassem Al Ismaeil, Marcos Damian Perez, Michel Poucet, Djamila Aouada. "Spacecraft Recognition Leveraging Knowledge of Space Environment: Simulator, Dataset, Competition Design, and Analysis". IEEE International Conference on Image Processing Challenges **ICIPC 2021**.

2. **Mohamed Adel Musallam**, Miguel Ortiz Del Castillo, Kassem Al Ismaeil, Marcos

Damian Perez, Djamila Aouada " Leveraging Temporal Information for 3D Trajectory Estimation of Space Objects ". Proceedings of the IEEE/CVF International Conference on Computer Vision Workshop **ICCVW 2021**.

3. **Mohamed Adel Musallam**, Miguel Ortiz del Castillo, Vincent Gaudillière, Kassem Al Ismaeil, Djamila Aouada. " CubeSat-CDT: A Cross-Domain Dataset for 6-DoF Trajectory Estimation of a Symmetric Spacecraft ". AI4Space workshop, proceedings of the 17th European Conference on Computer Vision. **ECCVW 2022** .

4. **Mohamed Adel Musallam**, Vincent Gaudillière, Miguel Ortiz del Castillo, Kassem Al Ismaeil, Djamila Aouada. "Leveraging Equivariant Features for Absolute Pose Regression". **CVPR 2022** - IEEE/CVF Conference on Computer Vision and Pattern Recognition, June 2022, New Orleans, USA.

5. **Mohamed Adel Musallam**, Vincent Gaudillière, Djamila Aouada. "Self-Supervised Learning for Place Representation Generalization across Appearance Changes". **WACV 2024** - IEEE/CVF Winter Conference on Applications of Computer Vision, Jan 2024, WAIKOLOA, HAWAII, USA.

6. **Mohamed Adel Musallam**, Vincent Gaudillière, Djamila Aouada. "SEPNet: Spacecraft Equivariant Pose Estimation Network". - Under Review.

PUBLICATIONS NOT INCLUDED IN THE THESIS

In the interest of conciseness and focus within this thesis, discussions about additional publications I have contributed are not included. This decision ensures a concise and targeted presentation of the core topics addressed.

1. Albert Garcia, **Mohamed Adel Musallam**, Enjie Ghorbel, Vincent Gaudillière, Marcos Perez, Kassem Al Ismaeil and Djamila Aouada. " LSPnet: a 2D localization-oriented spacecraft pose estimation neural network ". Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops **(CVPRW) 2021**.

2. Michele Jamrozik, Vincent Gaudillière, **Mohamed Adel Musallam**, Djamila Aouada. "Space Debris: Are Deep Learning-based Image Enhancements part of the Solution?" International Symposium on Computational Sensing **(ISCS) 2023**.

3. Vincent Gaudillière, Leo Pauly, Arunkumar Rathinam, Albert Garcia Sanchez, **Mohamed Adel Musallam**, Djamila Aouada. "3D-Aware Object Localization using Gaussian Implicit Occupancy Function " International Conference on Intelligent Robots and Systems **(IROS) 2023**.

4. Marcos D. Perez, **Mohamed Adel Musallam**, Albert Garcia, Enjie Ghorbel, Kassem Al Ismaeil, Djamila Aouada, Paul Le Henaff. "Detection & identification of on-orbit objects using machine learning" In **European conference on space debris 2021**.

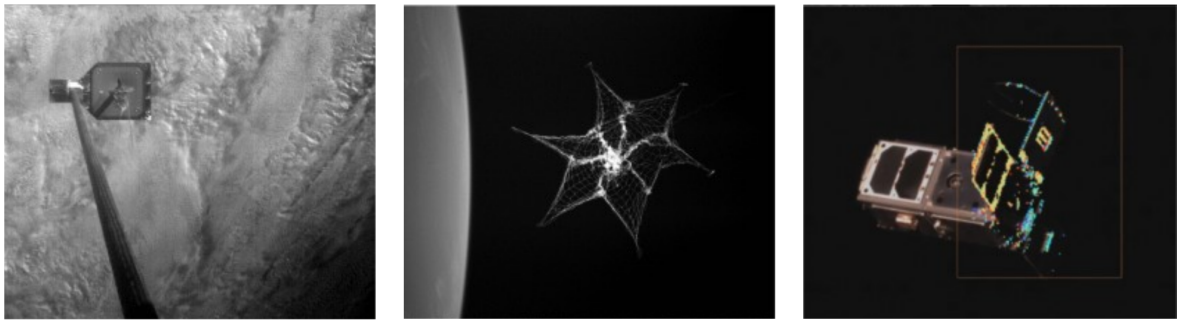
Chapter 2

Background

In this chapter, we lay down the theoretical and mathematical groundwork essential for understanding advanced computer vision techniques applied to space applications. We start with an introduction to SSA, ADR, and OOS, emphasizing the critical need for precise spacecraft pose estimation amidst the challenges of space imagery, such as variable lighting and sparse features.

We present the basics of pose estimation, focusing on the mathematical aspects of camera and object poses, with an emphasis on monocular vision methods. A review of Convolutional Neural Networks (CNNs) follows, highlighting their architecture and role in pose estimation, setting the stage for the advancements brought by Equivariant Group Convolutional Neural Networks (G-CNNs). The discussion extends to equivariant and invariant feature learning, underlining their significance in developing robust vision systems for the dynamic space environment. We introduce group theory concepts critical to G-CNNs, illustrating how they enhance traditional CNNs by incorporating transformation equivariance.

This chapter aims to provide a solid understanding of the concepts and methodologies that support the novel approaches discussed in this thesis, preparing the reader for a detailed exploration of solutions to the challenges of space exploration and debris monitoring.



(a) (b) (c)

Figure 2.1: (a) A CubeSat capture demonstration with Harpoon (b) Net targeting a CubeSat that was release from the main spacecraft (c) Vision-Based Navigation (VBN) system tracking the CubeSat. These technologies were integral to the Active Debris Removal demonstrations of the RemoveDebris mission [12].

2.1 Active Debris Removal and On-Orbit Servicing

Active Debris Removal (ADR) and On-Orbit Servicing (OOS) missions, at their core, involve the servicing or removal of non-operational space objects by an operational chaser spacecraft.

Figure 2.1 depicts images from RemoveDebris [12] mission the first global initiative to successfully test ADR technologies in orbit, showcasing cost-effective methods such as net capture and harpoon retrieval. It also demonstrated key components of the operation sequence, including vision-based navigation, marking a significant milestone in space debris management efforts.

This components as essential as they bridge to the more advanced stages of ADR missions, specifically the approach and capture/refueling phases. In these later stages, the focus shifts to managing the intricate relative dynamics between the chaser spacecraft and the target debris. These phases necessitate advanced Guidance, Navigation, and Control (GNC) techniques and highlight the necessity for autonomy, given the complex dynamics encountered, especially in close proximity. Thus, the integration of vision-based navigation within the RemoveDebris [12] mission exemplifies its essential value in enhancing the precision and autonomy of ADR missions, setting a foundational framework for future endeavors in space debris removal and management.

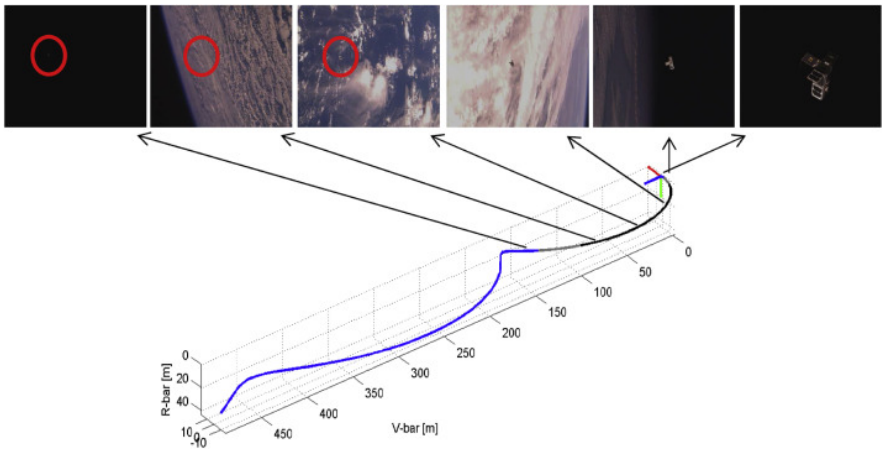


Figure 2.2: Tracking of DSAT2 within the VBN camera's field of view along the predefined trajectory, as illustrated in the figure from [12].

The evolution towards autonomous systems in forthcoming ADR/OOS missions is driven by the intricate and swift dynamic characteristics of space object removal and servicing tasks [10]. The approach phase, pivotal for the subsequent union of the spacecraft, highlights the limitations of latency-prone ground-based operations and the insufficient coverage of ground stations, thus underscoring the need for advancements in autonomous navigation. This includes accurately estimating the pose of a target object, a particularly demanding task given the typically uncooperative nature of targets in ADR/OOS missions, which do not facilitate navigation with aids like visual markers or LEDs [18].

Despite the absence of a universal benchmark for ADR/OOS mission navigation performance, historical and planned missions provide insights into expectations, particularly for the approach phase, which varies based on the mission's GNC system and objectives.

The RemoveDebris mission [12] features an experiment focused on vision-based navigation (VBN), described as follows [52]. In this setup, a second CubeSat, DSAT2, developed by Surrey Space Centre, is deployed from the platform at a minimal velocity. Subsequently, the VBN camera and LiDAR, a collaborative development by Airbus DS, CSEM, and Inria, gather data which is then transmitted back to Earth through the platform for analysis.

During the VBN experiment, the system on the platform monitored DSAT2, the designated CubeSat, executing various maneuvers as shown in Figure 2.2. These observations

spanned a range of distances and occurred under different lighting conditions that varied according to the spacecraft's orbit.

The CubeSat, designed as a 2U unit, incorporates avionics distributed within its frame, and features four deployable panels arranged in a cross shape at its base. These panels are designed purely to give the CubeSat a more satellite-like appearance, without serving any specific functional purpose.

The objectives of the VBN demonstration are multifaceted, aiming to:

- showcase advanced image processing and navigation algorithms, leveraging real flight data collected via two distinct yet complementary sensing devices: a conventional camera and a flash imaging LiDAR,
- conduct an in-flight validation of flash imaging LiDAR technology,
- implement an onboard processing capability to facilitate navigation tasks.

Thus the development of advanced VBN systems is critical for enhancing ADR and OOS missions. These systems provide essential capabilities for precise, real-time tracking and maneuvering, which are critical under the dynamic conditions of space. Accurate VBN enables chaser spacecraft to effectively interact with targets, navigating complex spatial environments with agility. The successful deployment and operation of VBN in the RemoveDebris mission underscores its vital role in supporting sustainable and safe space operations going forward.

2.2 Monocular Pose Estimation Methods for Spacecraft

In the context of on-orbit servicing and space rendezvous missions, precise pose estimation of a target spacecraft is a critical capability for a chaser spacecraft. The process involves a monocular vision system, which captures a two-dimensional image of the target then try to estimate the target's pose relative to the chaser's camera.

The process of observing a target spacecraft using a camera mounted on a chaser spacecraft is rooted in the principles of photogrammetry and computer vision. At its core is a

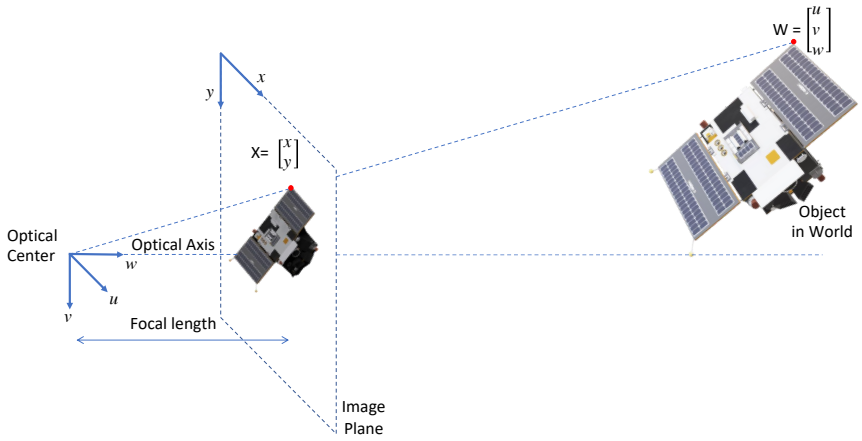


Figure 2.3: **Illustration of the Pinhole Camera Model in Spacecraft Imaging.** The optical center, akin to a pinhole, is located at the origin of the 3D camera coordinate system, labeled as (u, v, w). The image plane, the surface where the virtual image of the spacecraft is projected, is offset along the w-axis, referred to as the optical axis. The point of intersection where the optical axis meets the image plane is termed the principal point. The focal length is defined as the distance between the image plane and the optical center.

camera model, often conceptualized as a pinhole camera, which is essential for interpreting the 2D images captured in space.

2.2.1 Camera Model Fundamentals

In the camera model, the optical center of the camera, acting as the pinhole through which light passes, is positioned at the origin of the world coordinate system, denoted by the axes (u, v, w) as shown in Figure (2.3). The image plane, which is the two-dimensional surface where the image is formed, is placed parallel to the uv-plane of the camera coordinate system and is offset along the w-axis, also known as the optical axis.

2.2.2 Mathematical Representation

In the domain of spacecraft observation, the transformation from the three-dimensional space to the two-dimensional image plane is governed by the intrinsic parameters of the camera.

These parameters include the focal lengths along the x and y axes of the image sensor, represented by f_x and f_y. Additionally, the intrinsic parameters are characterized by the principal point, noted as (C_x, C_y), which is defined as the point where the optical axis intersects the image plane.

When a point X with 3D coordinates (x, y, z) is captured by the camera, it is projected onto the image plane as a point p with image coordinates (u, v). This projection is mathematically expressed by the camera's intrinsic matrix $\mathbf{K}$ through the following equations:

$$\mathbf{p} = \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \frac{x}{z} f_x + C_x \\ \frac{y}{z} f_y + C_y \end{bmatrix}.$$

The coordinates (u, v) on the image plane are thus computed using the focal lengths f_x, f_y, and the principal point coordinates (C_x, C_y).

To integrate the camera's perspective with the position and orientation of the target, homogeneous coordinates are employed. This system simplifies the equations and allows for a unified representation of points in space.

The relationship between a 3D point in the world coordinate system and its 2D image is encapsulated as follows:

$$\omega_i \begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} = [\mathbf{KP}] \begin{bmatrix} x_i^B \\ y_i^B \\ z_i^B \end{bmatrix},$$

where, ω_i is a scaling factor, and $[\mathbf{K}]$ represents the intrinsic matrix of the camera, which includes its optical properties. The matrix $[\mathbf{P}]$ is known as the pose matrix, which is composed of a rotation matrix $\mathbf{R}_B^C$ that aligns the target's body frame B with the camera frame C, and a translation vector $\mathbf{t}^C$ that represents the position of the target's center of mass in the camera frame.

The pose matrix $\mathbf{P} = [\mathbf{R}_B^C | \mathbf{t}^C]$ is crucial as it embodies the target's exact location and orientation from the viewpoint of the camera, enabling precise calculations needed for spacecraft navigation and maneuvering.

2.2.3 Camera's Role in Spacecraft Observation

The camera mounted on the chaser spacecraft serves as an eye in space. It captures the light reflected from the target spacecraft and forms a 2D representation of the 3D object on its image sensor. Through the camera's optics and its internal parameters, the raw data from the 3D environment is converted into a format that can be analyzed computationally.

The image plane is where the 2D image is formed and is characterized by its perpendicularity to the optical axis. The optical center is the point from which the distances to the image plane are measured, commonly referred to as the focal length. This focal length is a key factor in determining how the 3D scene is projected onto the 2D plane.

In summary, the camera model used in space missions is a critical component that translates the three-dimensional reality of space into two-dimensional images. Understanding this model is essential for various tasks in space exploration, including navigation, inspection, and docking procedures. The model provides the framework within which the observed data can be interpreted correctly to inform the decisions and maneuvers of the chaser spacecraft.

2.2.4 Determining the Target's Position and Orientation

Once the camera has captured an image of the target spacecraft, the next step is to understand precisely where the target is located and how it is oriented in relation to the camera. This information is critical for maneuvers such as docking or maintaining a safe distance.

Position of the Target with Respect to the Camera Frame

The position of the target's center of mass in the camera's coordinate system is denoted as $\mathbf{t}_C$. It represents the 3D spatial coordinates that define where the center of the target spacecraft is located from the perspective of the camera. This position is crucial for understanding the relative distance between the two spacecrafts.

Orientation with Respect to the Camera Frame

In addition to position, knowing the orientation of the target spacecraft is essential. This is represented by the rotation matrix $\mathbf{R}_{BC}$, which provides the transformation needed to align the target spacecraft's body frame B with the camera's frame C. This matrix encapsulates the angular displacement about each axis needed to rotate from the body frame of the target to that of the camera.

2.3 Convolutional Neural Networks

Convolutional Neural Networks (CNNs) have revolutionized the field of image processing and computer vision, becoming the cornerstone of numerous applications, including spacecraft pose estimation [23].

At their core, CNNs are designed to automatically learn spatial hierarchies of features from input images. The architecture of a CNN is structured in layers, with the convolutional layer being the central building block. These layers consist of a set of filters or kernels, which slide over the input image to produce feature maps through convolution operation.

This process, when applied to an image, involves sliding the kernel over the input image and computing the dot product at each position. The strength of convolution lies in its ability to capture local patterns within the input image efficiently, regardless of their location within the image. The intuition behind the convolution operator's success as a building block in deep learning architectures stems from its capacity to achieve two critical objectives: parameter sharing and sparse interactions. Parameter sharing reduces the model's complexity by using the same weights across different parts of the input, enabling the network to learn and generalize patterns more effectively. Sparse interactions focus the computation on local patches of the input data, making the learning process more efficient and scalable.

CNNs utilize the convolution operator, which is defined within the $\mathbb{R}^2$ domain for a image signal $I : \mathbb{R}^2 \to \mathbb{R}$ and a kernel $k : \mathbb{R}^2 \to \mathbb{R}$, applied at a point $\mathbf{x} \in \mathbb{R}^2$:

$$(I * k)(\mathbf{x}) = \int_{\mathbb{R}^2} I(\tilde{\mathbf{x}}) k(\tilde{\mathbf{x}} - \mathbf{x}) \mathrm{d}\tilde{\mathbf{x}}, \tag{2.1}$$

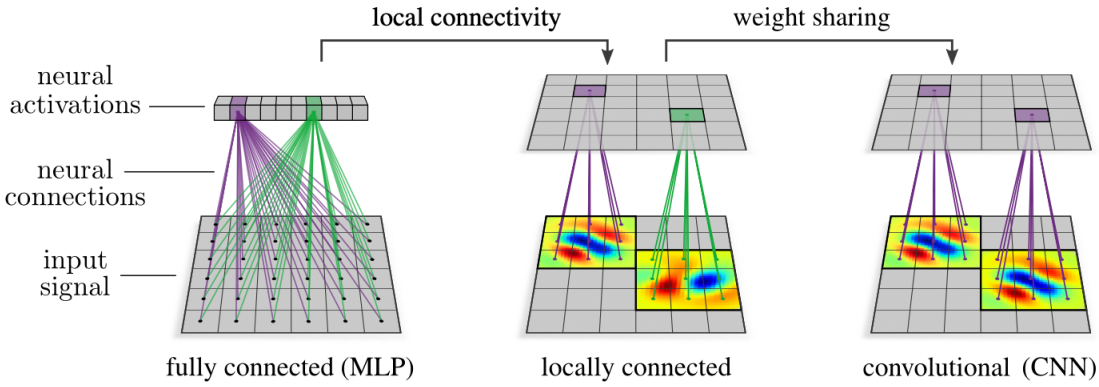


Figure 2.4: Evolution of neural network architectures from fully connected (left) to convolutional (right), demonstrating the principles of local connectivity and weight sharing. Fully connected (MLP) networks lack spatial structure, connecting every input signal to each neural activation. Locally connected networks introduce spatial structure by limiting connections to local regions. CNNs further refine this structure by applying weight sharing, using the same filters across the entire input field, which enhances pattern detection efficiency and consistency. Figure adapted from Weiler, Maurice *et al.* [53].

This equation describes the convolution as the inner product between the image signal I and a spatially shifted kernel k.

It is important to note that CNNs in practice utilize a discretized form of this operation:

$$\begin{aligned}(I * k)(\mathbf{x}) &= \sum_{\tilde{\mathbf{x}} \in \mathbb{Z}^2} I(\tilde{\mathbf{x}}) k(\mathbf{x} - \tilde{\mathbf{x}}) \Delta\tilde{\mathbf{x}} \\ &= \sum_{\tilde{\mathbf{x}} \in \mathbb{Z}^2} I(\tilde{\mathbf{x}}) k(\mathbf{x} - \tilde{\mathbf{x}})\end{aligned} \tag{2.2}$$

In the context of digital images, where pixels are uniformly spaced, we assume $\Delta\tilde{\mathbf{x}} = 1$ for simplicity in this discussion, even though our overview remains within the continuous domain. Typically, both I and k are composed of multiple channels, and the operation sums over these channels.

CNNs tend to learn patterns that exhibit a high degree of redundancy with respect to geometric transformations like rotations as some properties of images are stable under transformations. It's not uncommon to observe in the early layers of a CNN that the kernels are essentially rotated versions of one another—for instance, one kernel might serve as an

edge detector in one orientation, while another detects edges at a different angle [33]. The strength of CNNs lies in their capacity for weight sharing across various spatial locations in the data through the correlation operator, allowing a single kernel to be reused at every position. However, this weight sharing typically does not extend to rotations or orientations, although an examination of learned features suggests that incorporating such invariance could significantly reduce redundancy.

In many types of data, one can anticipate that fundamental patterns will manifest themselves under various rotations. This is true for 2D images, where patterns like edges, corners, and lines are ubiquitous. Introducing weight sharing across rotations and orientations in CNNs could, therefore, provide a more efficient way of learning these patterns, decreasing redundancy and potentially enhancing the network's overall performance [38].

2.3.1 Invariance and Equivariance

As we mentioned above, certain properties of images remain consistent despite undergoing transformations. To understand this concept in a mathematical framework:

A function $f(I)$ associated with an image I is said to be invariant to a transformation $T(I)$ if:

$$f(T(I)) = f(I). \tag{2.3}$$

This implies that the function's output $f(I)$ remains unaltered irrespective of the transformation applied to I. Image classification networks, for instance, should exhibit invariance to the geometric transformation of the image such as translations, rotations, flips, or other distortions. The network $f(.)$ is expected to recognize the same object within an image, even after it has been subjected to various spatial modifications.

Conversely, a function $f(.)$ is deemed equivariant or covariant to a transformation $T(.)$ if:

$$f(T(I)) = T(f(I)). \tag{2.4}$$

In essence, $f(.)$ is equivariant to the transformation $T(.)$ when its output undergoes the same transformation as the input itself. For tasks like per-pixel image segmentation, networks

should preserve this equivariance; if an image is translated, rotated, or flipped, the network $f(.)$ should generate a segmentation map that is similarly translated, rotated, or flipped.

Similarly, networks designed for pose estimation tasks must also demonstrate equivariance to the transformations of objects within the input images. This is crucial because as the position, orientation, or scale of an object changes within the visual field, the network's output—a prediction of the object's pose—must adjust accordingly in a consistent manner. Thus, if an object in an image is translated, rotated, or scaled, a well-constructed pose estimation network $f(.)$ should account for these transformations, ensuring that the estimated pose reflects these modifications precisely. Such equivariance is fundamental for accurate pose estimation, enabling reliable and robust recognition and analysis of objects' spatial orientations in diverse scenarios.

2.3.2 Group Convolutional Neural Networks

The adoption of convolutional layers in neural networks is motivated by their intrinsic property of approximate equivariance to translations, whereas the pooling layers aim to provide invariance to small translational shifts [54, 55].

However, when it comes to preserving precise spatial information through the network, there is a significant amount of spatial information regarding the inputs that are not encoded by CNNs in a precise fashion [31, 32]. More specifically, local and global poolings, if added to CNNs, render translation information unrecoverable, discarding the foregoing equivariance [56]. For this reason significant interest in developing networks capable of demonstrating equivariance or invariance to a wide array of transformations, extending well beyond mere translational changes. This includes manipulations such as reflections, rotations, and scaling, which are commonplace in varied imaging scenarios. In pursuit of this goal, Sifre & Mallat (2013) [57] innovated a framework grounded in wavelet transformations that ingrained translational and rotational invariance within image segments, showcasing its efficacy in the domain of texture classification. Furthermore, Kanazawa et al. (2014) [58] advanced the concept with the creation of convolutional neural networks that inherently possess local scale invariance.

Taking a more theoretical approach, Cohen & Welling (2016) [38] harnessed the principles of group theory to architect Group Convolutional Neural Networks (G-CNNs). These networks are adept at maintaining equivariance across an extended range of transformations, encapsulating reflections and rotations, thereby increasing the versatility of CNNs for complex tasks.

Group Theory Primer

To begin our overview of G-CNNs, we will present some foundational concepts in group theory. A group is characterized by the pair $(G, \cdot)$, where G represents a set of elements, and $\cdot$ denotes the binary operation that defines how these elements are combined. For a structure to be considered a group, the following criteria must be met:

1. Closure: For any two elements $g_1, g_2 \in G$, the operation $g_1 \cdot g_2$ results in another element within G.

2. Identity Element: There exists an element e in G such that for every $g \in G$, the operation $e \cdot g = g \cdot e = g$ holds true.

3. Inverses: For each element $g \in G$, there exists an inverse element $g^{-1} \in G$ such that $g \cdot g^{-1} = e$.

4. Associativity: The group operation is associative; for any $g_1, g_2, g_3 \in G$, the relation $(g_1 \cdot g_2) \cdot g_3 = g_1 \cdot (g_2 \cdot g_3)$ is always true.

Groups can perform actions on functions that are defined over $\mathbb{R}^2$ such as images, an operation that can be described using the regular representation, denoted as $\mathcal{L}_g^{\mathbb{G}\to\mathbb{R}^2}$. For brevity, we refer to this simply as $\mathcal{L}_g$. This operation is defined by the formula:

$$\mathcal{L}_g I(\mathbf{x}) = I\left(g^{-1} \cdot \mathbf{x}\right), \tag{2.5}$$

where the action of g^{-1} on $\mathbf{x}$ is represented as $g^{-1} \cdot \mathbf{x}$.

In the domain of G-CNNs, the goal is to achieve equivariance to a wider range of transformations $\mathbb{R}^2$. The objective is to construct a CNN that also exhibits equivariance to a

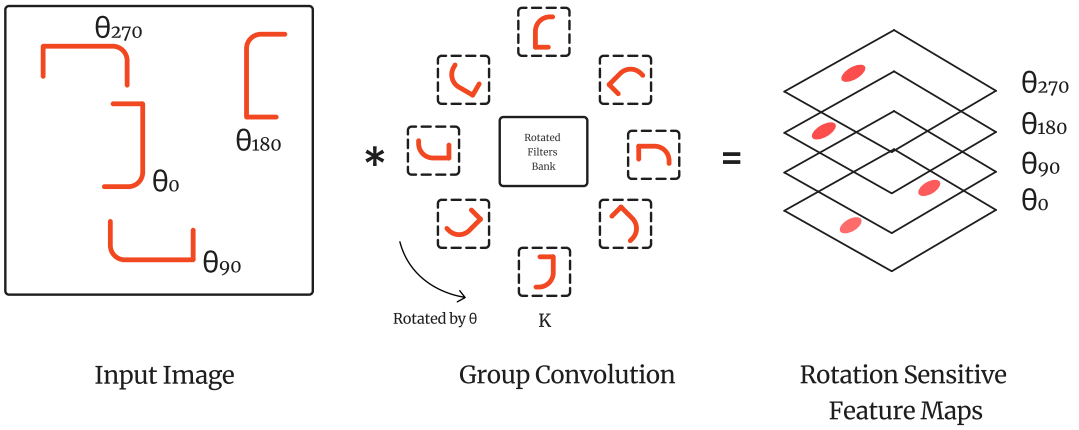


Figure 2.5: **Illustration of the lifting group convolution process**. Input image map I_{in} contains occurrences of a pattern e under various transformations, represented by θ_x with x is the angle of rotation. The group convolution operation transforms these features by matching them with a rotated versions of kernel K to produce an output feature map f_{out} that is extended along the group dimension $G = \mathbb{R}^2 \rtimes H$. The figure visualizes how features under different transformations are registered at distinct offsets in the group dimension, as exemplified with the group of 90° rotations, H. Figure reproduced from [61].

more comprehensive group of transformations, G, which typically encompasses translations in $\mathbb{R}^2$ and an additional group of interest, H. Here our discussions will primarily revolve around groups formed by combining translations with another group, focusing specifically on the Cyclic group of order 4, $\mathrm{H} = \mathrm{C}_4$, representing 90-degree rotations in 2D. For more in-depth analysis and information on equivariance across various groups in the context of Group G-CNNs, please consult [59, 60, 53].

To address 2D images defined on $\mathbb{R}^2$ effectively, the initial step towards developing a network capable of identifying the specific pose (or transformation from the group $G = \mathbb{R}^2 \rtimes H$) under which an input feature appears involves transforming our signal to a domain where the same feature, albeit in a different pose, is distinguished. This transformation is facilitated by the lifting convolution, which projects features from the input signal $I_{\text{in}} : \mathbb{R}^2 \to \mathbb{R}$ to a feature map defined on the group $I_{\text{out}} : G \to \mathbb{R}$. Considering a image signal and kernel I, k

both situated in $\mathbb{R}^2$, and a group element $g = (\boldsymbol{x}, h) \in G = \mathbb{R}^2 \rtimes H$:

$$(I *_{\text{lifting}} k)(g) = \int_{\mathbb{R}^2} I(\tilde{\mathbf{x}}) k_h(\tilde{\mathbf{x}} - \mathbf{x}) \mathrm{d}\tilde{\mathbf{x}},$$

where k_h represents the transformation of the kernel $k : \mathbb{R}^2 \to \mathbb{R}$ under the regular representation $\mathcal{L}_h$ of a group element $h \in H$; $k_h = \frac{1}{|h|}\mathcal{L}_h[k]$.

With the feature map now defined on the group $I_{\text{out}} : G \to \mathbb{R}$, we proceed to apply group convolutions. This process expands the convolution operation to include integration over the entire group G, as shown below:

$$\begin{aligned}(I *_{\text{group}} k)(g) &= \int_G I(\tilde{g}) k\left(g^{-1} \cdot \tilde{g}\right) \mathrm{d}\tilde{g} \\ &= \int_{\mathbb{R}^2} \int_H I(\tilde{\mathbf{x}}, \tilde{h}) \mathcal{L}_x \mathcal{L}_h k(\tilde{\mathbf{x}}, \tilde{h}) \frac{1}{|h|} \mathrm{d}\tilde{\mathbf{x}} \mathrm{d}\tilde{h} \\ &= \int_{\mathbb{R}^2} \int_H I(\tilde{\mathbf{x}}, \tilde{h}) k\left(h^{-1}(\tilde{\mathbf{x}} - \mathbf{x}), h^{-1} \cdot \tilde{h}\right) \frac{1}{|h|} \mathrm{d}\tilde{\mathbf{x}} \mathrm{d}\tilde{h}.\end{aligned} \tag{2.6}$$

This approach, termed group convolution, differs from the lifting convolution primarily in that both the signal and kernel I, k are now functions on the group G, extending the integral to span the entirety of G. Beyond this extension, the core concept remains largely consistent.

Following several layers of group convolution, we achieve a representation that remains equivariant to the actions of the group, and spatial information regarding the inputs are encoded in this new group representation. In Chapter 5, we will explore the application of G-CNNs in the context of pose estimation.

Chapter 3

Spacecraft Data Simulation and Collection

In this chapter, we detail our efforts in creating a simulation environment specifically designed for modeling and synthetic data generation. This work marks a substantial contribution to the field, notably through the organization of competitions and the distribution of public datasets for various spacecraft recognition tasks. We also delve into our investigation of the domain gap issue, a pivotal challenge in applying machine learning models to real-world scenarios.

Our exploration begins with the development of the SPARK simulation system, a pivotal tool for generating synthetic imagery essential for training machine learning models aimed at SSA. The creation and sharing of these datasets not only fuel research in spacecraft recognition but also provide a tangible framework for addressing the domain gap problem—a critical obstacle in the practical application of these models.

Following this, we present a comprehensive examination of the methodologies employed in constructing this simulation environment, highlighting the technical considerations and innovative approaches undertaken. We further explore the implications of the domain gap on the performance of machine learning models, offering insights into potential solutions and mitigation strategies.

This chapter sets the stage for an in-depth discussion on the intersection of simulation, synthetic data generation, and machine learning in the context of SSA and ADR. It aims to equip the reader with a thorough understanding of our contributions to this domain, paving the way for a deeper examination of specific challenges and solutions in subsequent chapters.

3.1 Introduction

The development of DL models for spacecraft pose estimation requires comprehensive training to adhere to the stringent performance criteria required for space applications. The efficacy of these models is intrinsically linked to the quality of the training datasets. Indeed, the dataset's quality is often as pivotal as the development of an efficient DL algorithm in attaining desired performance metrics. Training DL models typically involves utilizing extensive datasets, covering a wide array of application scenarios, which is crucial for enabling the models to generalize effectively across unencountered situations. Despite advancements in DL algorithms towards more efficient methodologies like few-shot and zero-shot learning, the challenge of accurately determining 6-DoF pose predominantly relies on large and diverse datasets.

A major impediment in the broader application and validation of DL models in this domain is the scarcity of publicly accessible space-borne image datasets. This limitation is especially pronounced in cases where space-borne images of specific targets are either unavailable or restricted. In response, image rendering tools have become a favored approach for generating realistic space-borne images. These tools, along with on-ground validation testbeds, facilitate the creation of thousands of annotated images for varied applications, including object detection, semantic segmentation, and 6 DoF pose estimation. These generation tools offer substantial adaptability, allowing for the modification of parameters such as camera models and orbital lighting conditions to suit specific application requirements.

In spacecraft pose estimation applications, these algorithms are typically integrated into vision-based navigation systems and validated in specialized testbed facilities. These facilities are designed to simulate orbital relative motion, employing mechanisms like robotic arms or

air-bearing platforms, under realistic space lighting conditions. Depending on the application's requirements and the facility's constraints, target mock-ups used in these testbeds can vary in scale. While synthetic imagery can be produced in large volumes to meet diverse needs, the generation of imagery in testbed scenarios is more constrained, owing to the need to replicate specific conditions like Earth's background, the sun's precise position, and Earth's albedo, which differentiate lab/testbed imagery from actual space imagery.

This scenario presents three distinct image domains in spacecraft pose estimation: synthetic, laboratory, and actual space imagery, each relevant in the development, testing/validation, and deployment stages. A notable challenge in this field is the "domain gap" problem, where DL models tend to overfit to features specific to the training domain. Addressing domain generalization from a data perspective is therefore critical to enhance the performance of these algorithms in spacecraft pose estimation.

In this chapter, we detail our efforts in building a simulation environment tailored for modeling and synthetic data generation. This work represents a significant contribution to the field, particularly in the organization of challenges and sharing public datasets for various spacecraft recognition tasks. Additionally, we explore into our investigation of the domain gap problem, a critical issue in the application of machine learning models to real-world scenarios.

3.2 SPARK Simulation

Simulation software tools are essential in the contemporary design of vision-based autonomous systems, particularly in the realm of spacecraft engineering.

The escalating complexity of spacecraft missions and maneuver designs necessitates the heavy reliance on software simulation tools for dynamic and kinematic design verification, as well as post-launch telemetry analysis.

These tools enable engineers to enhance design quality and testing efficiency, thereby reducing both cost and duration of development. Additionally, the growing need for vision-based perception and navigation in spacecraft necessitates the development of realistic sim-

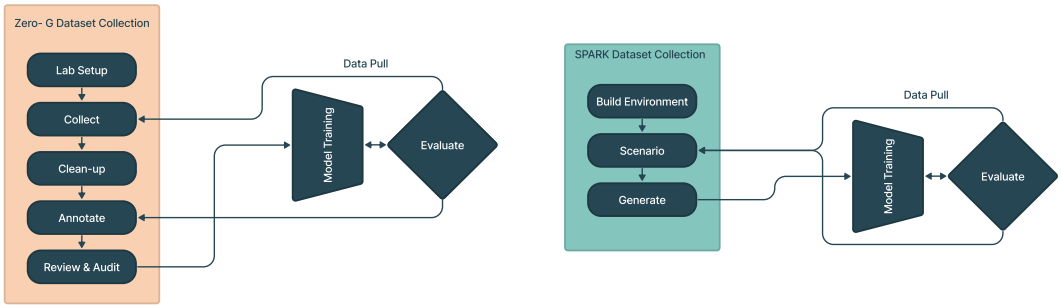


Figure 3.1: **Comparison of Dataset Iteration Processes**: Real-World vs. Synthetic. On the left, the Zero-G lab dataset illustrates necessary steps like collection, cleanup, annotation, review, and audit, involving significant human effort and time. On the right, the SPARK-generated synthetic dataset process showcases the construction of a 3D asset-based environment, adjustment of randomization parameters, and environment execution to produce new data. This method yields precisely annotated datasets that are auto-validated, substantially reducing time-intensive tasks.

ulations for the generation of synthetic data.

To address this need, we have developed SPARK, a robust astrodynamics framework. This framework provides simulation of spacecraft's vision-based close rendezvous and proximity operations under conditions akin to space.

Astrodynamics simulation tools can generally be categorized into three types: Commercial off-the-shelf (COTS), Government off-the-shelf (GOTS), and general open source. Notably, several tools originated in the GOTS category before transitioning to open source.

The following is a list of prominent tools:

1. University of Surrey's STAR LAB (URSO) [62]
2. Stanford's Space Rendezvous Lab (SPEED) [63].
3. University of Colorado Autonomous Vehicle Systems Lab (BASILISK) [64].
4. AGI STK [65].
5. a.i. FreeFlyer [66].
6. ESA PANGU Simulator [67].

7. NASA General Mission Analysis Tool (GMAT) [68].
8. NASA Trick [69].
9. OreKit [70].
10. Airbus SurRender [71].
11. DART/Dshell [72]

Each tool is designed with specific astrodynamics simulation objectives in mind. For instance, tools like OreKit, GMAT, and STK initially focused on high-fidelity orbit dynamics, estimation, propagation, and trajectory design. ESA PANGU and Airbus SurRender were developed for modeling planetary bodies and asteroid surfaces.

Consequently, these tools encompass various propagators, intricate multi-body gravity models, drag, solar radiation pressure, and orbit determination tools. Our primary aim in developing SPARK is to facilitate the development of machine learning models for vision-based autonomous navigation of spacecraft.

SPARK incorporates space-like lighting and environmental elements to deliver a realistic and precise representation of spacecraft operations and their responses to various space conditions.

Its advanced features enable detailed visualization and analysis of spacecraft behavior, taking into account the need for fast developing and validation for machine learning development cycle as shown inf Figure (3.1). This enhanced functionality renders SPARK suitable for in developing vision models for OOS/ADR missions.

SPARK was developed using Unity3D [73], a versatile cross-platform game development engine. Game development platforms like Unity have become increasingly accessible to the public, for developing 3D environment for different application such as Virtual Reality (VR), Virtual Design and Construction (VDC) or Simulation.

Unity3D offers high-performance, drag-and-drop components such as cameras, Graphical User Interface (GUI) objects, lighting, and shaders. These elements are not only reusable but can also be customized to achieve high-end graphics rendering, a crucial aspect in accurately

simulating space environments. The ability to import spacecraft and component 3D models in various formats further enhances the flexibility and adaptability of the platform, allowing for extensive customization in line with specific project needs.

One of the significant advantages of using a platform like Unity3D is its extensive cross-platform support. It caters to a wide range of operating systems including Mac, Linux, Windows, iOS, and Android. This broad compatibility ensures that applications developed within Unity can be easily published across different platforms with minimal hassle. The simplicity of publishing an application to the desired platform with just a click significantly streamlines the development process, making it more efficient and user-friendly.

The use of Unity3D in developing SPARK underscores the engine's adaptability, high performance, and ease of use in creating complex simulations, such as those required for spaceborne systems. Its wide-ranging compatibility and user-friendly interface make it a suitable choice for developing sophisticated and visually rich applications in the realm of space exploration and beyond.

Generating synthetic data and ground truth labels using SPARK simulator is achieved by integrating a 3D model of the target spacecraft, a comprehensive 3D environment comprising the Sun, Earth, and Deep Space, along with the camera model mounted on the chaser spacecraft as illustrated in Figure (3.2). Additionally, the desired simulation can be achieved using either the Unity Perception Package [74] or a custom simulation script for more control.

This approach facilitates the acquisition of high-resolution, photorealistic RGB images, 3D pose labels, depth maps, and corresponding segmentation masks across diverse and varying environmental conditions as show in Figure (3.3).

The fidelity of spaceborne imaging is subject to a multitude of factors, including fluctuating illumination conditions, signal-to-noise ratio, and high contrast levels. To address this, the SPARK generated data has been meticulously curated to encompass a broad spectrum of scenarios, inclusive of extreme and challenging conditions. The generated images spans multiple axes:

1. **Scene Illumination**: The generated images models the impact of Sun flares, rays, and reflections from Earth in space, mirroring diverse illumination conditions and the pro-

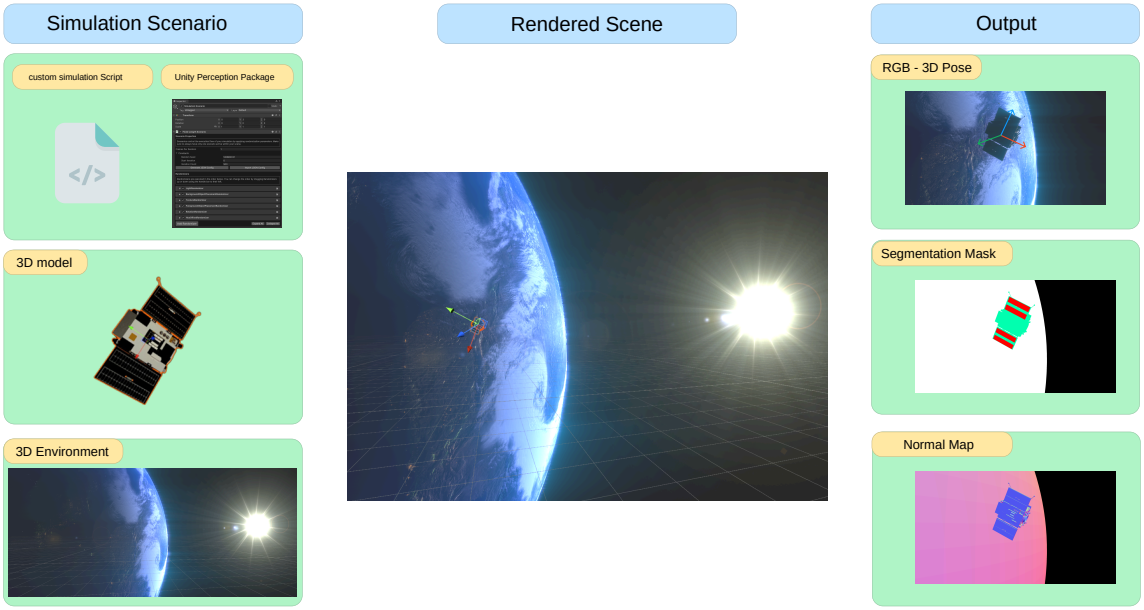


Figure 3.2: **Workflow of a SPARK simulation**: The left column demonstrates the simulation scenario components including simulation scripts and a 3D model within a 3D environment setup. The middle column depicts the rendered scene resulting from the simulation, showcasing the satellite model in orbit around Earth with the sun in the background. The right column presents the output data generated from the simulation: an RGB image with a 3D pose overlay, a segmentation mask for object identification, and a normal map for surface analysis

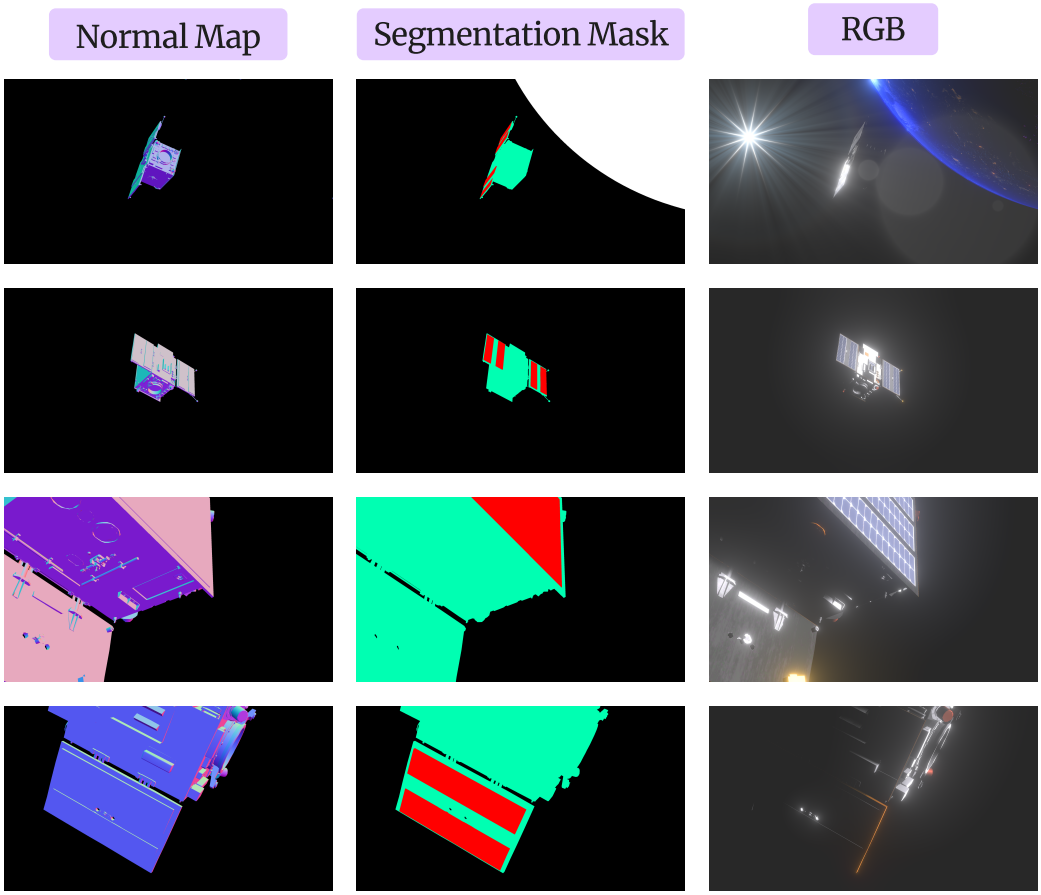


Figure 3.3: **Visualization of Proba-2 spacecraft using SPRARK**: This figure illustrates three types of data outputs from the SPARK simulator. The Normal Map column provides detailed surface normal, the Segmentation Mask column offers object segmentation for scene understanding, with the solar panel labeled with different mask in red, and the RGB column displays the realistic rendering of the Proba-2 spacecraft in space with different illumination effects

nounced contrast characteristic of spaceborne images. It includes extreme illumination scenarios where sunlight directly impacts optical navigation sensors or reflects off the target surface or Earth, causing lens flare and optical sensor blooming effects.

2. **Scene Background**: The orientation of the target spacecraft in various orbital scenarios can either face Earth or the dark expanse of space, leading to variable backgrounds. Earth as a background introduces additional features such as the planet's surface, and high reflectivity from oceans and clouds. Conversely, a dark space background results in a featureless backdrop with sparsely illuminated stars.

3. **Distance between Camera and Target**: The model simulates varying distances between the target spacecraft and the optical sensor on the chaser spacecraft. The range is inversely proportional to the target's occupation percentage in the scene.

4. **Optical Sensor Noise**: To realistically simulate the high noise levels prevalent in spaceborne images, attributable to small sensor sizes and high dynamic range imaging, varying levels of zero-mean white Gaussian noise have been added to the synthetic images. This addition replicates the noise observed in actual spaceborne images.

In summary, the SPARK simulator's comprehensive approach to synthetic data generation ensures a robust and realistic representation of spaceborne imaging conditions, crucial for the development and testing of vision-based autonomous navigation systems for spacecraft.

3.3 Zero-G Facility

The SnT Zero-Gravity Laboratory (Zero-G Lab) at the University of Luxembourg [75] facility represents a state-of-the-art facility designed to replicate a broad array of in-orbit operations across various orbital environments.

Equipped with two sophisticated Universal Robots (model UR10e) robotic arms, each mounted atop a Cobotrack rail system, the lab offers a flexible $6 + 1$ Degrees of Freedom (DoF) configuration. These robotic arms are instrumental in precisely emulating the orbital paths of spacecraft, additional space objects, and light sources, which are crucial for authentic

experimental setups. To accurately simulate the intricate lighting conditions encountered in space, the lab is equipped with a high-quality Godox SL-60 LED Video Light.

In terms of interior design, the Zero-G Lab, spanning an area of 3×5 meters, is intentionally configured to resemble space-like conditions, with black walls, ceiling, and epoxy flooring, effectively minimizing light reflections. The experimental configuration includes several vital components, as illustrated in Figure (3.4). These include cameras positioned on tripods facing the spacecraft, typically a model spaceraft, mounted on a UR10e robotic arm, and a dark backdrop strategically placed behind the spaceraft. Additionally, a continuous light source is mounted on a second robotic arm. This arrangement is versatile, accommodating a variety of spacecraft mock-ups and carefully considering room dimensions, reflective surfaces, camera constraints, and the robotic arms' range of motion. facilitates comprehensive simulation of final approach maneuvers - refer to Figure (3.5)-.

During experimental procedures, the camera is stationary while the robotic arms adjust the positions of the spacecraft and the light source. The arms' movements are guided by manually programmed waypoints, with Python scripts utilized to automate the image capture process. These scripts are crucial for adjusting camera settings to ensure accurate white balance and prevent color shifts. The positioning of background materials and cameras is methodically executed to precisely replicate space conditions, with the background set approximately 56 cm behind the spacecraft's center and the cameras placed about 140 cm in front.

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.

Object movements within the lab are tracked using an OptiTrack system with six Primex 13W cameras, capable of operating at up to 240 Hz with invisible 850nm IR illumination. These cameras track IR markers attached to the objects, yielding highly accurate pose labels

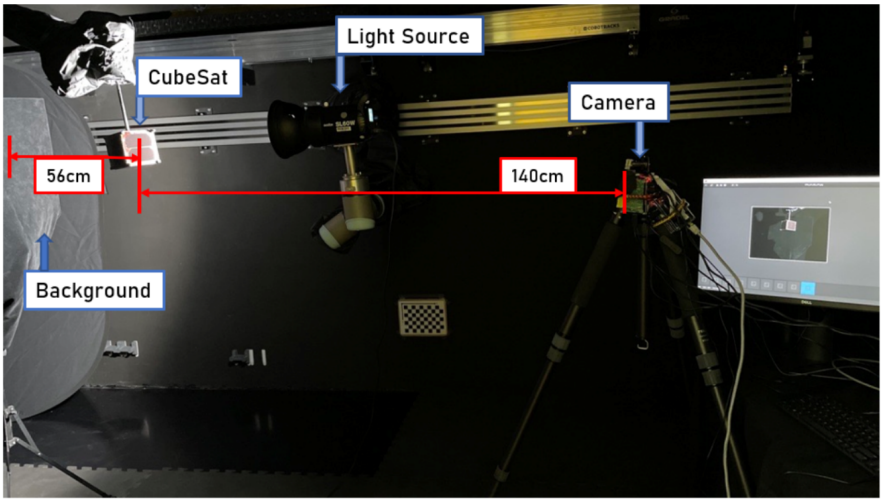


Figure 3.4: **Overview of the SnT Zero-G Lab's laboratory** configuration for data acquisition. The setup includes a CubeSat as the spacecraft model and a light source, both attached to movable UR10e robotic arms. Cameras are stationed on a stable tripod, with backgrounds positioned strategically behind the CubeSat.

with minimal positional and rotational errors.

3.4 Dataset Generated using SPARK

The *SPARK Challenge 2021*, acronym for "SPAcecraft Recognition leveraging Knowledge of Space Environment," was an event organized by the Computer Vision, Imaging & Machine Intelligence Research Group (CVI²) at the IEEE International Conference on Image Processing (ICIP). This challenge was aimed at fostering the development of data-driven approaches for space target recognition, and motivating researchers from computer vision and machine learning field to tackle challenges in the new domain of spacecraft recognition.

A central feature of the SPARK Challenge was the introduction of a new, unique space multi-modal annotated image dataset. This dataset was significant in its scale and specificity, consisting of approximately 150,000 high-resolution, photorealistic RGB images. Each image in this collection came with bounding box annotations identifying the target object. Additionally, the dataset included around 150,000 depth images and an equivalent number of segmentation masks, all representing various and diverse space environmental conditions.

The dataset's composition was designed to reflect a realistic representation of space scenarios. It encompassed images of 10 different satellite types, with each satellite type represented by approximately 12,500 images. Furthermore, the dataset included images of 5 different debris objects, with each object represented by about 5,000 images. All debris objects were collectively categorized into a single class.

This rich and varied dataset provided a resource for researchers and practitioners in the field of space exploration and technology. By offering a wide array of annotated images depicting satellites and space debris under different conditions, the *SPARK Challenge 2021* set the stage for advancements in the field of space object recognition, particularly in the application of machine learning and computer vision techniques. The challenge was not only a testament to the growing need for sophisticated space object detection and classification methods but also highlighted the potential of collaborative and interdisciplinary research initiatives in addressing complex challenges in space technology.

Target spacecraft: Ten different realistic models of spacecrafts were used (*'AcrimSat', 'Aquarius', 'Aura', 'Calipso', 'CloudSat', 'Jason', 'Terra', 'TRMM', 'Sentinel-6'*, and the *'1RU Generic CubeSat'*). They were obtained from NASA 3D resources [76]. The debris are parts of satellites and rockets after adding corrupted texture to simulate dysfunction conditions (*'space shuttle external tank', 'orbital docking system', 'damaged communication dish', 'thermal protection tiles',* and *'connector ring'*). They were placed around the Earth and within the low Earth orbit (LEO) altitude.

Chaser spacecraft: It represents the observer equipped with different vision sensors used to acquire data.

Camera: A pinhole camera model was used with known intrinsic camera parameters and optical sensor specifications, as well as a depth camera for range imaging.

Following the success of the *SPARK 2021 challenge*, a second iteration of the challenge, named SPARK2022, was organized. This subsequent challenge was a part of the AI4Space workshop, held in conjunction with the European Conference on Computer Vision (ECCV) in 2022.

The *SPARK 2022 Challenge* was focused on advancing the development of data-driven

approaches specifically for spacecraft detection and trajectory estimation. This represented a shift towards more dynamic aspects of space object tracking and analysis, expanding the scope from the previous challenge which concentrated primarily on recognition and classification.

A notable aspect of the *SPARK 2022 Challenge* was its use of data from two distinct sources. Firstly, it utilized data synthetically simulated using our SPARK simulator. This approach allowed for the creation of highly realistic and varied datasets, simulating a range of space conditions and scenarios that are essential for developing robust detection and trajectory estimation algorithms.

Secondly, the challenge incorporated data collected from the Zero-G Lab facility [75]. This real-world data provided a complement to the synthetically generated data. The Zero-G Lab, offered unique insights into the behavior of spacecraft and objects in space-like environments. The combination of synthetic and real-world data in the SPARK2022 Challenge provided a comprehensive dataset for participants, offering a more holistic and challenging environment for developing and testing their algorithms.

The inclusion of trajectory estimation in the challenge underscored the evolving needs in spacecraft monitoring and management, particularly in the context of increasing space traffic and the consequent need for precise navigation and collision avoidance systems. The SPARK2022 Challenge, with its integration of both synthetic and real-world data, highlighted the critical issue of the domain gap between real and synthetic data. This aspect of the challenge underscored the need for further research in bridging these differences to enhance the effectiveness of machine learning models in practical, real-world applications in space environments.

3.5 SPARK-T Dataset

To further study of spacecraft trajectory estimation, we utilized our SPARK simulator to generated a wide array of diversity in terms of sensing conditions and spacecraft trajectories.

Developing the *SPARK-T* dataset, we employed 3D models of three distinct types of spacecraft:

1. The 'Jason' satellite, represented by a 3D model with dimensions of $3.8m \times 10m \times 2m$, including deployed solar panels.

2. A 1RU generic 'CubeSat', with dimensions of $10cm \times 11cm \times 11cm$.

3. For representing space debris, a model of a heat shield tile measuring $15cm \times 10cm \times 3cm$ was used.

These 3D models were sourced from NASA's [76] extensive library of 3D resources.

In creating the *SPARK-T* dataset, the target spacecrafts were strategically placed in various trajectories within the camera's field of view, which was mounted on a chaser spacecraft. Additionally, to enhance the realism and variability of the simulated space environment, both the Sun and Earth were animated to rotate around their respective axes. This approach ensured the generation of a diverse set of high-resolution, photorealistic RGB images, encompassing a range of different orbital scenarios.

For this project, 50 sequences were created for each of the three spacecraft types. Each sequence comprised 50 frames, capturing the spacecraft's 3D trajectory. The dataset included ground truth data for these trajectories, along with the corresponding rotation (R) and translation (t) parameters of the spacecraft relative to the camera's reference frame.

Finally, to mimic the imaging conditions of space, all images were resized to a resolution of 512×512 pixels. They were then processed with a zero-mean Gaussian blurring, with a variance of $\sigma^2 = 1$, and an added Gaussian white noise with a variance of $\sigma^2 = 0.001$. This processing technique was crucial in simulating the noise characteristics typically observed in spaceborne imaging, thereby enhancing the dataset's utility for developing and testing trajectory estimation algorithms.

The comprehensive details regarding the development of the model are thoroughly presented in Chapter (7), provides an in-depth exploration of the methodologies, techniques, and considerations involved in the model's creation. It aims for understanding the intricacies of the development process, offering insights into both the theoretical underpinnings and practical applications of the model.

3.6 CubeSat-CDT dataset

To inspect the challenge of trajectory estimation using data from different sources and to provide a more detailed study on the issue of domain gap, we introduced the CubeSat Cross-Domain Trajectory (CDT) dataset. This dataset is unique in its composition, combining a variety of data sources to offer a comprehensive view of spacecraft trajectory estimation.

The CubeSat CDT dataset includes:

1. 21 trajectories of an actual CubeSat, which were captured in a laboratory setup. This real-world data provides an authentic baseline for trajectory behavior.

2. 50 trajectories generated using SPARK, a powerful game development platform renowned for its realistic rendering capabilities. These synthetic trajectories, created in a controlled virtual environment, offer high-quality data for comparison against real-world scenarios.

3. 15 trajectories generated using Blender, a comprehensive open-source 3D creation suite. Blender's trajectories provide an alternative set of synthetic data, contributing to the dataset's diversity.

In total, the CubeSat CDT dataset comprises approximately 22,000 high-quality and high-resolution images. These images depict a 1U CubeSat moving along predefined trajectories, offering a rich resource for analyzing and understanding trajectory patterns in both real and simulated environments.

To provide a detailed comparison and analysis, Table (3.1) in the relevant document outlines the minimum and maximum distances between the CubeSat and the camera, categorized according to the data domain. This tabulation is crucial for understanding the variations in data characteristics across different sources and for studying the impact of these variations on trajectory estimation models. The inclusion of both real and synthetic data in this dataset allows for addressing the domain gap challenge, offering insights into how models perform across varied data sources and how they might be optimized for improved accuracy and reliability in real-world applications.

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

Table 3.1: Minimum and maximum distances between the CubeSat and the camera for each data domain of the proposed CubeSat CDT dataset.

3.6.1 Zero-G Laboratory Data

For the acquisition of real images in the Zero-G Lab, we conducted in lab simulations that emulate the 6D dynamic motion of two satellites during an orbital rendezvous. This process was crucial in creating a realistic and practical dataset for our research.

The specific trajectories defined for the CubeSat CDT in these simulations were choosen to replicate close-range operations typically encountered when a 1U CubeSat is deployed in orbit.

The resultant dataset comprises 21 trajectories of an actual CubeSat, all captured within the controlled environment of the Zero-G Lab. This collection of real-image trajectories provides invaluable data for the development and testing of algorithms in spacecraft recognition and pose estimation, offering a practical perspective on the dynamics of satellite movements in an orbital setting.

3.6.2 SPARK Synthetic Data

The first subset of synthetic data within the CubeSat Cross-Domain Trajectory (CDT) dataset, known as *SPARK Synthetic Data*, was generated using the SPARK simulator. This simulator was specifically designed to produce visuals that closely mimic real-world scenarios of a target model in space. Key features of this subset include:

1. **Configurable Lighting Conditions**: The SPARK simulator allowed for the creation of a range of lighting environments, enabling the simulation of various space lighting scenarios. This feature is crucial for testing the robustness of trajectory estimation algorithms under different illumination conditions.

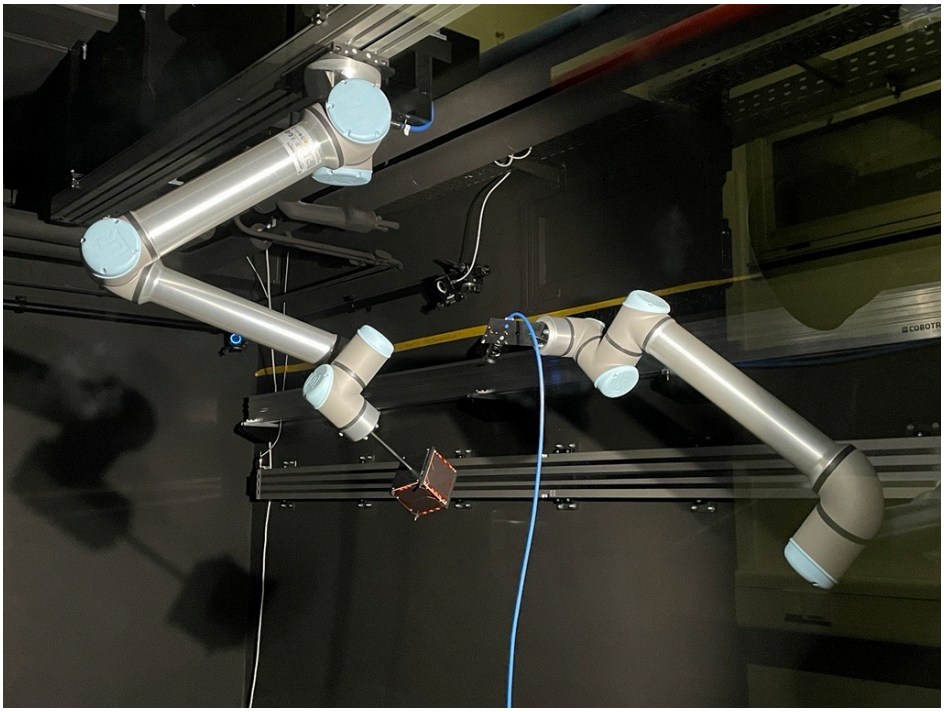


Figure 3.5: **Experimental Setup for Orbital Rendezvous Simulation**: This image depicts the in-lab setup used for simulating the 6D dynamic motion of two satellites during an orbital rendezvous in the Zero-G Lab [75].

2. **Earth Background Option**: The simulator provided the flexibility to include or exclude an Earth background in the visual data. This variation adds to the realism of the dataset and challenges the algorithms to perform accurately in visually complex scenarios.

3. **Predefined Trajectories with a Fixed Camera**: In this setup, the virtual target (the CubeSat model) was programmed to follow predefined trajectories, while the camera position remained fixed. This approach focuses on the movement patterns of the target, providing clear data for trajectory analysis.

4. **Camera Intrinsic Parameters Matching Zero-G Lab Data**: To ensure consistency and relevance, the camera's intrinsic parameters in the SPARK Synthetic Data were aligned with those used in the Zero-G Lab dataset. This alignment is vital for

creating synthetic data that is comparable to real-world data, thereby enhancing the effectiveness of cross-domain studies.

The SPARK Synthetic Data subset is instrumental in bridging the gap between real and synthetic data, offering an environment for testing and refining trajectory estimation models. By simulating realistic space conditions and maintaining consistency with real-world data parameters, this subset plays a critical role in advancing the study of spacecraft trajectory estimation in diverse and challenging conditions.

3.6.3 Blender Synthetic Data

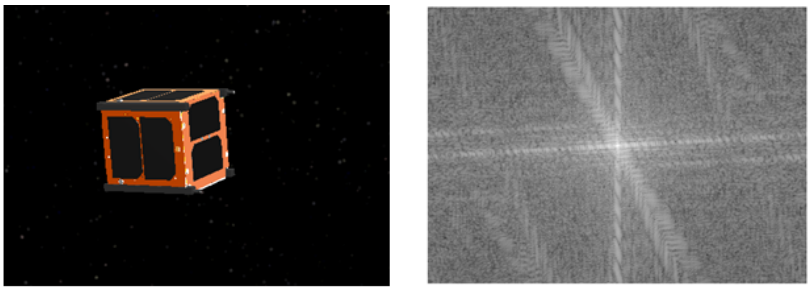
To increase the diversity of the dataset, we incorporated Blender-based synthetic data to add a significant layer of diversity to the dataset. Blender [77], a renowned open-source 3D computer graphics software, offers unique capabilities and features for data generation. To create Blender synthetic data, we followed the following guidelines :

1. **Importing CubeSat CAD Model**: The Computer Aided Design (CAD) model of the CubeSat was imported into Blender. This step ensures that the 3D model used in Blender is a precise and accurate representation of the CubeSat, essential for generating realistic images.

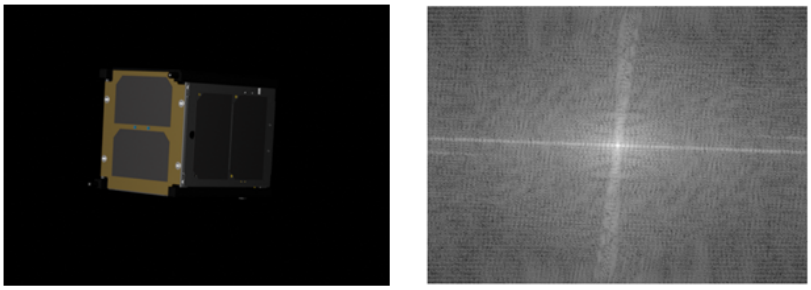
2. **Matching Camera Intrinsic Parameters**: To align the synthetic data with real-world conditions, the intrinsic parameters of the camera in Blender were adjusted to match the physical properties of the camera used in the Zero-G Lab. This alignment is crucial for consistency and allows for a direct comparison between synthetic and real datasets.

3. **Using Groundtruth Labels from Zero-G Lab**: The pose information from datasets generated in the Zero-G Lab was utilized as ground truth labels in the Blender rendering process. This approach ensures that the synthetic data closely mirrors the real-world scenarios, providing a reliable basis for model testing and validation.

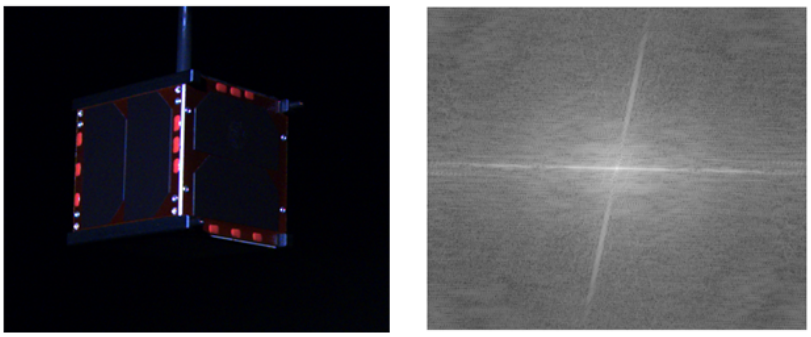
4. **Camera and Target Positioning**: In the Blender setup, the camera is fixed at the



(a) Synthetic (SPARK)



(b) Synthetic (Blender)



(c) Zero-G Lab

Figure 3.6: **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.

origin, and the target (CubeSat) is moved relative to the camera based on the pose information. This configuration simplifies the image acquisition process and facilitates the verification of ground truth labels post-rendering.

By using Blender in conjunction with the SPARK simulator, the dataset benefits from the strengths of two different rendering engines, each contributing its own unique qualities. This methodological diversity enhances the overall robustness and applicability of the dataset, making it a more comprehensive resource for developing and testing trajectory estimation models under a variety of simulated conditions.

Chapter (8) presents a detailed overview of the development and performance evaluation of models across various datasets, focusing on methodologies and techniques used in their creation. It emphasizes understanding the domain gap issue — the challenge of applying deep learning models trained on one type of data (like synthetic data) to different data sources (such as real-world data).

Chapter 4

Spacecraft Recognition Leveraging Knowledge of Space Environment

This chapter presents our contribution to the first edition of the *SPAcecraft Recognition leveraging Knowledge of space environment* (SPARK) competition.

This event, organized by our team in conjunction with the 2021 IEEE International Conference in Image Processing (ICIP 2021), was an attempt in fostering research and innovation in the field of space target recognition and detection.

The cornerstone of the SPARK competition was its unique synthetic dataset, comprising 150,000 annotated multi-modal images. We designed the dataset specifically to challenge and stimulate researchers to develop groundbreaking solutions in the realm of space target recognition.

The chapter provides an overview of the sturcture of the challange, our construction of the dataset, elaborating on its composition, the modalities of the images included, and the annotations provided. It also offers a comprehensive analysis of the outcomes derived from the 17 submissions received for the competition. This analysis includes insights into the methodologies adopted by the participants, the performance of their proposed solutions, and an evaluation of how these contributions advance the field.

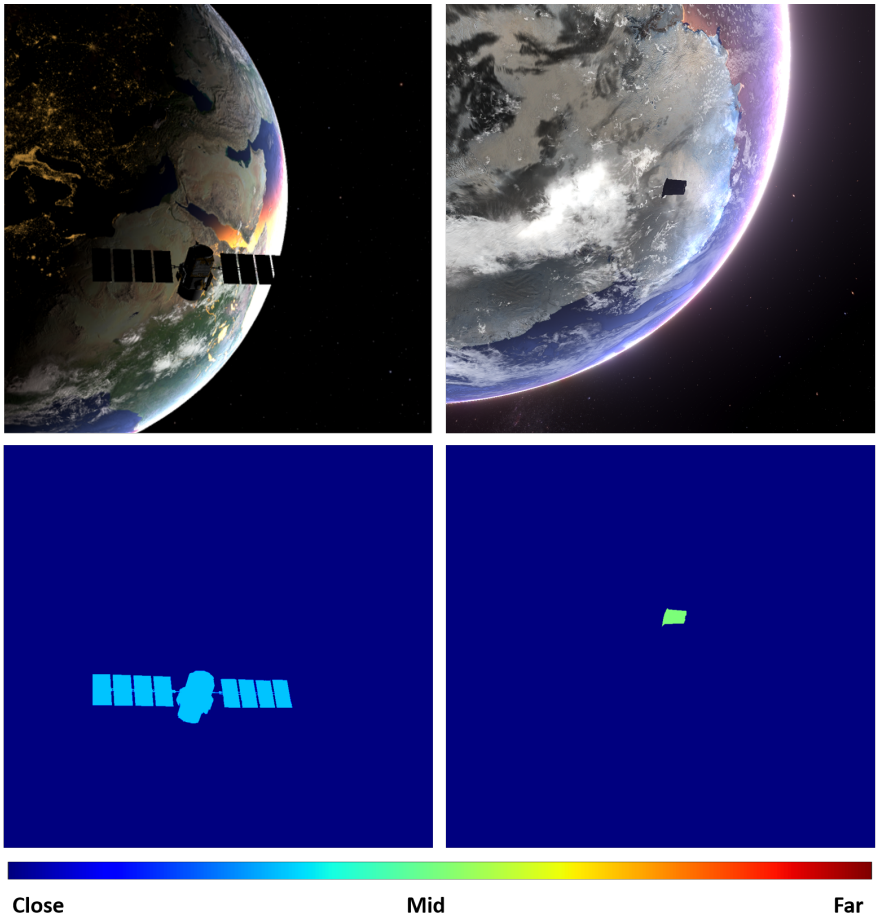


Figure 4.1: **Samples from our *SPARK* dataset**. Top-left: RGB image of the *'Calipso'* satellite with night side of Earth in the background. Top-right: RGB image of a debris with day side of Earth in the background. Bottom: corresponding depths.

4.1 Introduction

Today, our daily existence is deeply intertwined with the expansive satellite infrastructure that orbits our planet. This infrastructure plays a pivotal role in an array of services that are fundamental to our contemporary way of life, including communication, transportation, and weather forecasting, among others. Given this critical dependence, the protection of these space assets becomes paramount.

One of the primary hazards facing this infrastructure is the potential for collisions in space. Such events can have far-reaching consequences, not only damaging valuable equipment but

also potentially creating debris that can pose further risks. In this context, equipping spacecraft with the capability for autonomous recognition of surrounding objects emerges as a vital component of space safety measures. This capability is a key objective of SSA, aiming to significantly mitigate collision risks.

SSA encompasses a broad spectrum of activities, but at its core, it involves the monitoring and understanding of the space environment, including the detection, tracking, and cataloging of objects in orbit. The development of technologies and systems that enable spacecraft to autonomously detect and respond to nearby objects is crucial for maintaining the integrity and functionality of our satellite infrastructure, thereby safeguarding the myriad services that rely on it.

Objects of interest in SSA encompass a diverse array of entities, including both active and inactive satellites, as well as space debris. Space debris, in particular, poses a significant challenge due to its unpredictable nature and potential to cause harm to operational satellites and other space assets.

In recent years, the utilization of image-based sensors has gained considerable momentum as a crucial source of information for SSA. These sensors, capable of capturing visual data of objects in space, have become instrumental in tracking and monitoring the myriad of objects orbiting the Earth. The rich data provided by these sensors has, in turn, spurred a multitude of research initiatives in the field [10, 78, 9, 79, 15]. These research efforts are primarily focused on developing advanced algorithms and techniques for object detection, tracking, and classification using image data. The goal is to enhance the accuracy and reliability of identifying and categorizing various space objects, from operational satellites to fragments of debris. The advancements in image processing and machine learning, particularly in deep learning, have significantly contributed to these developments, offering new ways to interpret and utilize the data captured by image-based sensors.

The progress in this area is not just a technological pursuit; it has profound implications for the safety and sustainability of space operations. By improving our ability to monitor and understand the space environment, these research efforts play a crucial role in ensuring the longevity and security of our satellite infrastructure, which is vital for a wide range of

services and applications on which modern society depends.

Recently, there has been a significant surge in research exploring the potential of Deep Learning (DL) applied to image data for space applications. This burgeoning interest is largely due to the advancements in DL techniques, which have shown remarkable success in various fields, particularly in image processing and analysis [80, 62].

In the context of space applications, DL models are being investigated for their ability to effectively interpret and analyze images captured by space-based sensors. These models are trained to perform a range of tasks crucial for SSA and other space operations, such as:

1. **Object Detection and Classification**: Identifying and categorizing different objects in space, such as operational satellites, defunct satellites, and space debris.

2. **Pose Estimation**: Determining the orientation and position of space objects, which is vital for maneuvering and rendezvous operations.

3. **Anomaly Detection**: Identifying unexpected or abnormal patterns or objects, which could indicate potential hazards or system malfunctions.

4. **Change Detection**: Monitoring and detecting changes in the space environment over time, which is important for long-term space mission planning and safety.

The potential of DL in these areas is vast, offering improved accuracy, automation, and efficiency compared to traditional methods. For instance, DL models can process and analyze vast amounts of image data more quickly and accurately than human operators, providing timely insights critical for decision-making in space missions.

However, the application of DL in space environments also presents unique challenges, such as dealing with the vast and variable nature of space data, the limited availability of labeled training data, and the need for models to be robust against the unique noise and distortions present in space imagery.

Recent advancements in the field of SSA have led to the development of several synthetic and laboratory-acquired datasets, each designed to address specific challenges in space object analysis using Deep Learning (DL) techniques. These datasets include:

1. **Spacecraft Pose Estimation Dataset (SPEED) Dataset**: As referenced in [81] and [82], SPEED is tailored for 6D pose estimation, providing critical data for determining the orientation and position of spacecraft in space.

2. **Unreal Rendered Spacecraft On-Orbit (URSO) Dataset**: Cited in [62], the URSO dataset offers synthetic data also aimed at facilitating research in 6D pose estimation.

3. **Swiss Cube Dataset**: Mentioned in [83], this dataset contributes to the same field of 6D pose estimation, enriching the diversity of data available for research.

4. **Spacecraft-Parts Dataset**: Published in [84], this dataset diverges from the focus on pose estimation, instead concentrating on semantic segmentation, a crucial aspect for understanding the component parts of space objects.

While these datasets have significantly advanced research in pose estimation and semantic segmentation, there remains a notable gap in datasets specifically designed for space target recognition and detection. These tasks are fundamental for SSA and are critical for achieving autonomous operations in space.

Considering the substantial progress in object recognition using DL, as discussed in [85], there is a growing interest in exploring how these methods can be adapted and applied to space data. The unique challenges of the space environment, such as the lack of extensive labeled data, the varied and complex nature of space objects, and the specific lighting and background conditions in space, necessitate tailored approaches in DL.

To address these challenges and advance the field of SSA, there is a need for dedicated research efforts to create comprehensive datasets for space target recognition and detection. Such datasets would not only facilitate the application of existing DL techniques to space data but also spur innovation in developing new methods that are better suited to the unique requirements of space environments. The development of these datasets and subsequent research in applying DL to space data will be crucial steps towards enhancing the capabilities and autonomy of space missions.

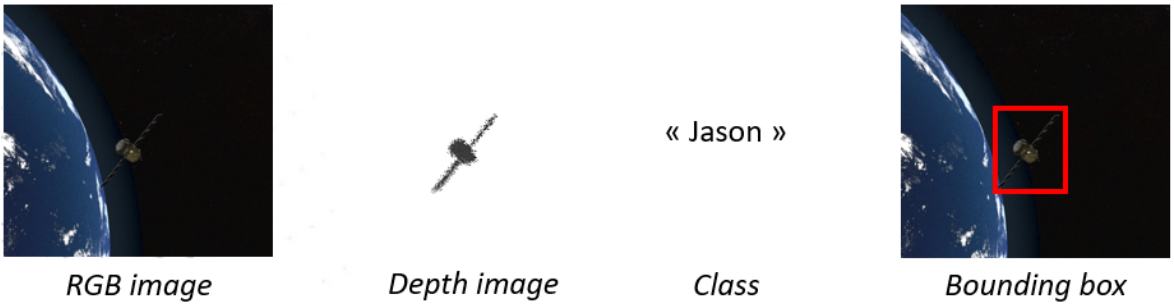


Figure 4.2: **Illustration from the SPARK Challenge Dataset**.

The introduction of the SPARK competition, which offers a new public space annotated image dataset dedicated to space target recognition and detection, arrives at a crucial juncture in the advancement of SSA. This dataset, accessible at [1] and illustrated in (4.2), is particularly significant when compared to previously available datasets for several reasons:

1. **Dual Modalities**: The SPARK dataset includes both RGB and depth images, providing a richer and more comprehensive data source for deep learning models.

2. **Volume of Data**: With a 150,000 RGB images and an equal number of depth images, the SPARK dataset significantly surpasses other datasets in size, the largest of which previously capped at around 21,000 images. This vast collection of data offers unprecedented opportunities for training more robust and accurate models.

3. **Diverse Object Classes**: The dataset encompasses 11 distinct object classes, including 10 different types of spacecraft and a separate class for space debris. This diversity in object classes is crucial for developing versatile recognition and detection algorithms.

4. **Photo-Realistic Space Simulation**: The images in the SPARK dataset were generated under a photo-realistic space simulation environment, ensuring a high level of realism. This includes a wide variety of sensing conditions, some of which are extreme and pose significant challenges for object recognition algorithms, the SPARK simulation is presented in section (3.2).

[1] https://cvi2.uni.lu/about-spark-2021/

Dataset	SPARK (Ours)	SPEED [81]	URSO [62]	Spacecraft-Parts [84]	SwissCube [83]
# images per modality	**150 k**	15 k	15 k	2.5 k	21 k
# Spacecrafts	**15**	1	2	-	1
Visible	✓	✓	✓	✓	✓
Color	✓	✗	✓	✓	✓
Depth	✓	✗	✗	✗	✗
Mask	✓	✗	✗	✓	✓
3D Model	✓	✗	✗	✗	✓
6D Pose	✓	✓	✓	✗	✓
Rendering	**simulated**	simulated + lab	simulated	artists' views	simulated

Table 4.1: Comparison of Space Situational Awareness datasets.

5. **Inclusion of 6D Pose Annotations**: Similar to some of the previous datasets, the SPARK dataset includes 6D pose annotations, adding another layer of utility for more comprehensive space object analysis.

Figure (4.1) presents sample images from the SPARK dataset, effectively demonstrates the dataset's quality and diversity. These images serve as a visual testament to the dataset's comprehensive nature, showcasing a range of scenarios and conditions that are crucial for robust space target recognition and detection.

Further enhancing the understanding of the SPARK dataset's significance, Table (4.1) provides a detailed comparative analysis with existing benchmarks. This comparison elucidates the unique features and advantages of the SPARK dataset, such as its extensive size, diverse modalities, and wide range of object classes. Such detailed contextualization helps in appreciating the dataset's contribution to advancing research in space target recognition and detection.

The introduction of the SPARK dataset indeed signifies a major leap forward in the field of SSA. It provides researchers and developers with an invaluable resource to develop and test advanced systems for identifying and tracking objects in space. By offering a more diverse and extensive set of data, it lays the groundwork for creating more sophisticated and capable recognition and detection technologies, thereby enhancing the safety and sustainability of space operations.

In this chapter, a comprehensive presentation of the SPARK dataset and the methodology employed in the SPARK competition is provided. This includes an in-depth overview of the dataset's structure, the variety of image modalities it encompasses, and the specific challenges

it poses for space target recognition and detection. The methodology section details the competition's framework, including the criteria for participation, the evaluation metrics used, and the overall objectives and goals of the competition.

Additionally, the chapter includes an aggregate analysis of the submissions received for the competition. This analysis offers insights into the various approaches and techniques employed by participants, highlighting the innovative methods and strategies used to tackle the challenges presented by the SPARK dataset. The results of the submissions are reported, summarizing their performance in terms of accuracy, efficiency, and their adherence to the competition's guidelines and objectives.

By analyzing and reporting on the submissions in aggregate form, the chapter provides a valuable overview of the state of research in the field of space target recognition and detection. It identifies common trends, challenges, and areas of success within the submissions, offering a snapshot of the advancements and potential areas for further research and development in this domain. The chapter thus serves as a significant resource for researchers and practitioners in the field, offering a comprehensive understanding of the SPARK competition's outcomes and contributions to the field of SSA.

Section (3.4) in Chapter (3) describes the proposed SPARK dataset. The details related to the challenge are given in Section (4.2) including the analysis of the submissions.

4.2 Challenge

The SPARK dataset was introduced to the academic community through a Challenge Session at the 2021 IEEE International Conference on Image Processing[2]. This session provided a platform for showcasing the dataset's features and potential, fostering collaboration and discussion among image processing experts and researchers.

[2]https://www.2021.ieeeicip.org

Figure 4.3: The SPARK Challenge Session at the 2021 IEEE International Conference

4.2.1 Competition Design

The primary goal of the Challenge was to examine the potential advantages of using multimodal data in solving two essential tasks in SSA: target object recognition and in-image localization. These tasks are pivotal for the accurate identification and tracking of objects in space, an area of increasing importance given the growing density of space traffic.

1. **Target Object Recognition (Classification Task)**: This task focuses on the ability to accurately classify various objects in space, such as satellites or space debris. It involves identifying the type or category of an object based on its visual characteristics. In this document, this task is referred to as the 'classification' task, emphasizing its focus on categorizing objects into predefined classes.

2. **In-Image Localization (Detection Task)**: The 'detection' task, on the other hand, involves determining the precise location of these objects within an image. This task is crucial for determining the position and potentially the trajectory of space objects, which is vital for collision avoidance and mission planning.

The Challenge sought to assess how effectively multimodal data, including RGB and depth images, can improve performance in these two tasks. Multimodal data offers a richer, more comprehensive dataset than unimodal data, potentially providing more nuanced information for algorithms to analyze. The use of such data in deep learning models could lead to more

accurate and reliable recognition and localization of space objects.

By focusing on these tasks, the Challenge aimed to advance the field of SSA by encouraging the development and testing of new methods and models

Experimental Setup

In the SPARK competition, participants were provided with a dataset comprising 150,000 RGB images, each accompanied by a corresponding simulated depth map, as illustrated in Figure (4.2). This dataset was divided into three subsets: 90,000 images for training, 30,000 for validation, and another 30,000 for testing purposes. The dataset features 15 different space objects, including 10 different satellites and 5 types of space debris.

For the classification task, the five debris types were grouped into a single class labeled as *'debris'*, while each satellite type was designated as an individual class. The primary objective for participants in this task was to accurately classify the type of spacecraft depicted in each pair of RGB and depth images.

In addition to classification, the competition also included a detection task. This task required participants not only to correctly identify the spacecraft but also to accurately localize it within the image by defining its bounding box. It's important to note that the ground truth data, consisting of spacecraft class labels and rectangular bounding boxes, was only released for the training and validation subsets, as shown in Figure (4.2). This approach was designed to test the models' abilities to generalize and perform accurately on unseen data, a crucial aspect of SSA applications.

Metrics

To evaluate the different submissions and rank the participants, two metrics were specifically designed.

Task 1 - Classification

In the classification task of the SPARK competition, participants faced two primary types of errors, each with varying degrees of seriousness:

1. Classifying Debris as a Satellite: This error type is considered more serious as the goal is to detect debris.

2. Misclassifying a Satellite: This could either be misidentifying a satellite as another type of satellite or as debris. This type of error is deemed less serious.

The overarching goals of the task were to:

1. Avoid confusion between satellites and debris, with a higher penalty applied for misclassified debris.

2. Accurately classify images featuring satellites based on their specific type.

To effectively rank participants while accounting for these error types and goals, a specific metric was designed:

$$\text{Perf} = F_2\text{-score}(debris) + \text{accuracy}(satellites) \tag{4.1}$$

In this metric:

$$F_2\text{-score} = 5\frac{\text{precision.recall}}{4.\text{precision} + \text{recall}} \tag{4.2}$$

The F_2-score is utilized to weigh recall (penalizing the omission of debris) as twice as important as precision (penalizing false debris detection). This scoring system is reflective of the task's emphasis on correctly identifying debris, a critical aspect of SSA.

Accuracy is defined as the proportion of correctly classified samples and is averaged over the 10 satellite classes. By adding the F_2-score for debris and accuracy for satellites, the metric balances the importance of both subproblems: avoiding misclassification of debris and correctly identifying satellite types. This approach ensures a comprehensive assessment of participants' performance in tackling the key challenges of the classification task.

Task 2 - Detection

For the detection task in the SPARK competition, the evaluation encompassed both the accuracy of localization and the performance of classification. The metric for this task was

significantly influenced by the evaluation framework used in the COCO Challenge. The specific approach adopted for the SPARK competition involved the following steps:

1. Classification Accuracy: Initially, the proportion of images correctly classified in terms of their target object (satellite or debris) is calculated.

2. Intersection-over-Union (IoU) Score:For each correctly classified image, the IoU score is computed. This score is a measure of the overlap between the predicted bounding box and the ground truth bounding box. An IoU threshold is set to determine whether the localization of the object is sufficiently accurate.

3. Averaging over IoU Thresholds: The final detection metric is calculated by averaging the proportions of correctly predicted images (in terms of both classification and localization) over a range of IoU thresholds. These thresholds vary from 0.5 to 0.95, increasing in increments of 0.05.

This comprehensive metric ensures that for a prediction to be deemed successful, it must not only correctly identify the type of object in the image but also accurately localize it within the image. By averaging over a range of IoU thresholds, the metric provides a robust and nuanced assessment of each submission's detection capabilities, reflecting both the precision of the bounding box localization and the accuracy of the object classification.

4.2.2 Analysis of Competition Results

In this section, a comprehensive analysis of the submissions for the SPARK challenge is presented, focusing on the overarching trends and insights rather than specific rankings.

Participation Overview:

The challenge saw participation from 5 teams, resulting in a total of 17 submissions. Each team submitted a report detailing their approach, providing a basis for the analysis presented here.

A key finding from the submissions is the apparent benefit of incorporating depth information for both classification and detection tasks. Teams employed depth data in two primary ways:

1. **As a Preprocessing Step**: Some teams used depth images to initially localize the satellite or debris within the image, followed by cropping the relevant section for further analysis.

2. **Fusion with RGB Images**: Other teams integrated depth information with RGB images during the training phase, enhancing the dataset's dimensionality and richness.

Data Augmentation Techniques:

Many teams employed data augmentation techniques to enhance model performance. These techniques included horizontal and vertical flips, Gaussian noise and blur, and more advanced methods like CutOut, MixUp, CutMix, and Mosaic.

Model Architectures:

A variety of well-known pre-trained model architectures were utilized, including ResNets [86], Efficient Det [87] and YOLO [88] models. These choices reflect the teams' reliance on established, robust architectures for complex image recognition tasks.

Classification Task Analysis:

The average results of submissions, represented in a confusion matrix Figure (4.4), indicate that correct classifications ranged from 54% to 81% across different classes. Notably, misclassifications often involved misidentifying satellites as debris, aligning with the Challenge's focus on accurately distinguishing these categories.

Detection Task Analysis:

For detection, the per-class results were analyzed against varying IoU thresholds Figure (4.5). As expected, higher IoU thresholds correlated with lower detection scores. Interestingly, the 'CloudSat' object emerged as particularly challenging in both classification and detection, warranting further investigation to understand this trend.

4.3 Conclusion

In conclusion, the SPARK Challenge, organized in conjunction with ICIP 2021, has successfully showcased the potential of the SPARK dataset in advancing the field of spacecraft

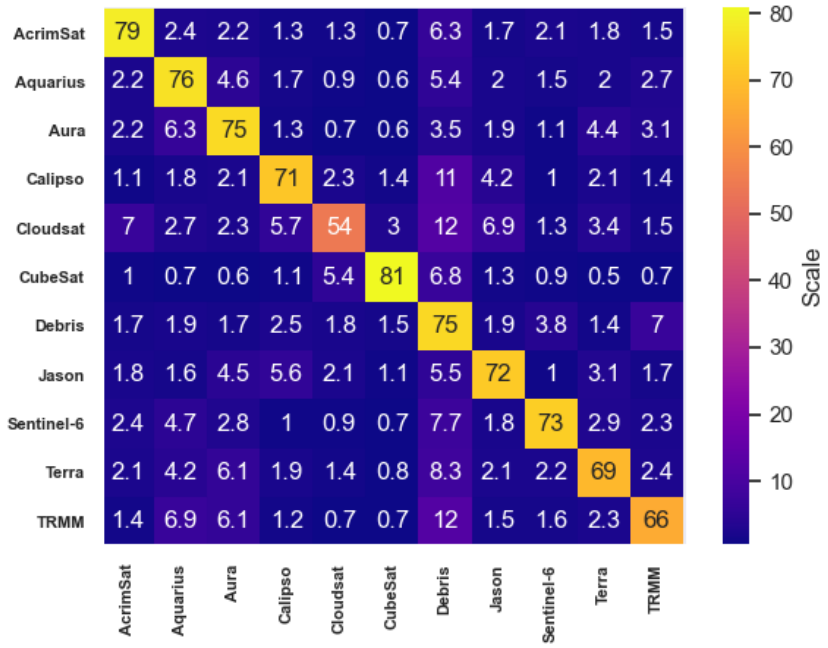


Figure 4.4: **SPARK Challenge 2021 (Task 1 – Classification)**: overall confusion matrix.

recognition and Space SSA. Our work in developing this dataset and orchestrating the Challenge has underscored the significance of multimodal deep learning approaches in this domain. The enthusiastic participation and interest from the global research community reaffirm the importance and relevance of multi-modal RGB-Depth data in enhancing spacecraft perception and detection capabilities. This endeavor not only contributes to the progression of SSA missions but also sets a new benchmark for future research in this vital area of space exploration and security.

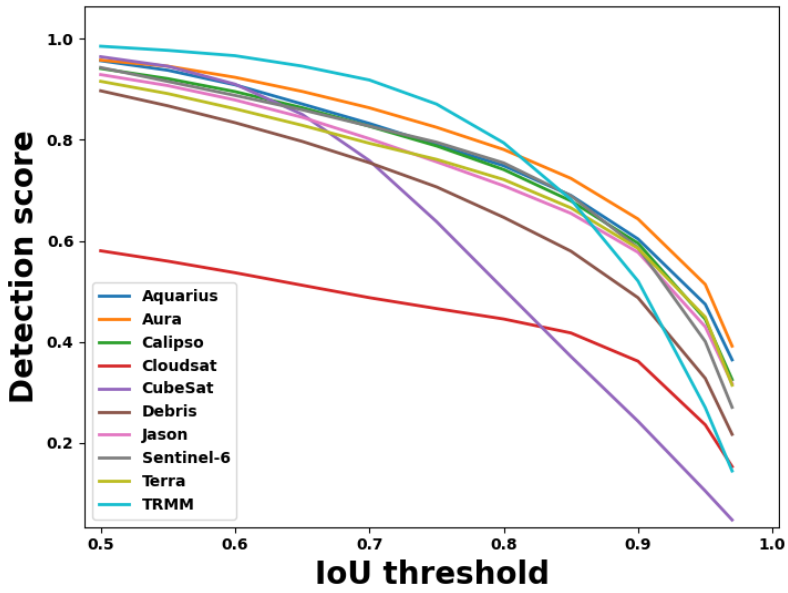


Figure 4.5: **SPARK Challenge 2021 (Task 2 - Detection)**: Proportion of correctly predicted images per class, as a function of the IoU threshold.

Chapter 5

Equivariant Features for Absolute Pose Regression

In this chapter, we explore the intricate problem of end-to-end relative pose estimation, a critical aspect for determining the position and attitude of a target spacecraft in relation to a chaser spacecraft. Our discussion primarily revolves around a theoretical investigation with broader applications in general camera pose estimation. The subsequent chapter will then shift focus specifically to spacecraft pose estimation.

While end-to-end approaches have achieved state-of-the-art performance in many perception tasks, they are not yet able to compete with 3D geometry-based methods in pose estimation. Moreover, absolute pose regression has been shown to be more related to image retrieval [89]. As a result, we hypothesize that the statistical features learned by classical Convolutional Neural Networks do not carry enough geometric information to reliably solve this inherently geometric task. In this chapter, we demonstrate how a translation and rotation equivariant Convolutional Neural Network directly induces representations of camera motions into the feature space. We then show that this geometric property allows for implicitly augmenting the training data under a whole group of image plane-preserving transformations. Therefore, we argue that directly learning equivariant features is preferable than learning data-intensive intermediate representations. Comprehensive experimental validation

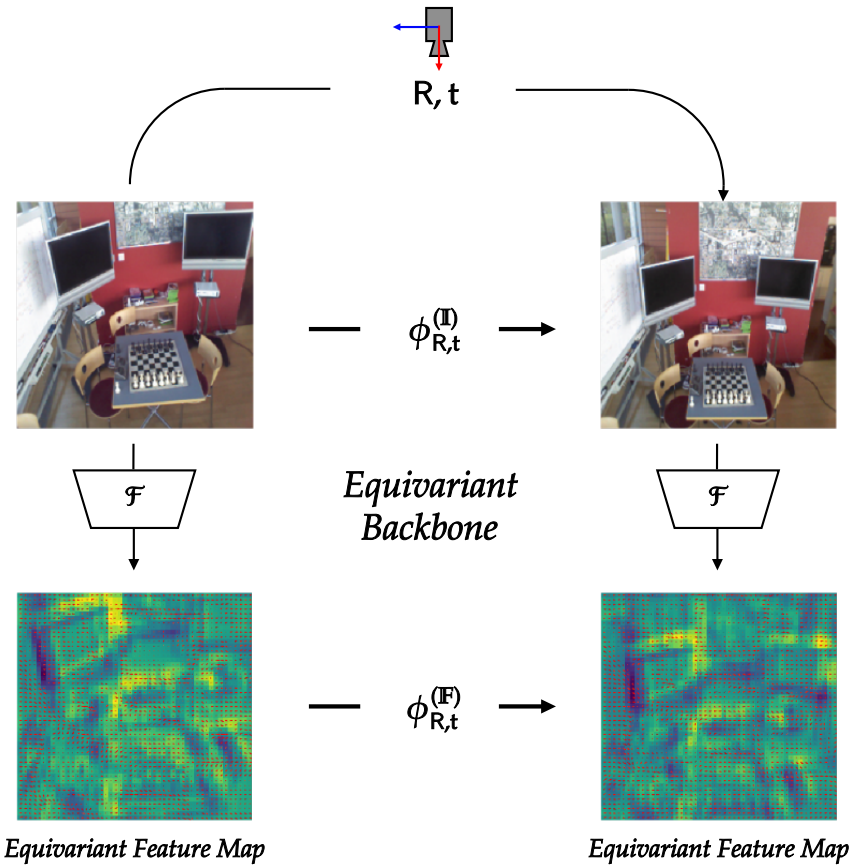


Figure 5.1: **Illustration of our approach** - Our method adopts a translation and rotation-equivariant convolutional neural network to extract geometry-aware features that directly encode camera planar motions $\mathsf{R}, \mathbf{t}$. While camera moves, equivariance of the proposed feature extractor $\mathcal{F}$ leads to explicit image ($\phi^{(\mathbb{I})}_{\mathsf{R},\mathbf{t}}$) and feature ($\phi^{(\mathbb{F})}_{\mathsf{R},\mathbf{t}}$) changes. This property is leveraged to propose a more efficient solution to the absolute pose regression problem.

demonstrates that our lightweight model outperforms existing ones on standard datasets.

5.1 Introduction

Estimating the relative pose between a camera and its environment is crucial for vision-based applications such as autonomous driving, robot manipulation, mixed reality and computer-assisted surgery. As a result, camera pose estimation, and its reference frame inverse, *i.e.*,

object pose estimation, have been extensively studied over the last decades [90, 91, 92].

Traditionally, pose estimation has been addressed using 3D geometry. In practice, a set of 2D-3D feature correspondences is generated, then statistically leveraged to recover the camera pose [93, 94, 95, 96]. More recently, direct Absolute Pose Regression (APR) approaches have been introduced, drawing upon early successes of deep learning [97]. These methods consist in directly mapping an image to its pose using a suitably trained Convolutional Neural Network (CNN). Therefore, end-to-end trainable methods have the advantage of providing fully differentiable results, enabling the optimization of all parameters in a comprehensive manner. Moreover, predictions are achieved at a steady speed and power consumption, whereas RANdom SAmple Consensus (RANSAC)-based methods [93] are less predictable and likely to suffer from an efficiency drop when the inlier rate is low. However, state-of-the-art APR methods have been proven theoretically and shown experimentally to have a lower accuracy compared to 3D structure-based approaches [89]. Indeed, the former are more closely related to image retrieval than to 3D structure [89].

The questions we ask in this work are: *Why do current APR methods fall short in accuracy ? How can they reach their full potential ?* Our hypothesis is that there is a lack of exploitation of the geometric properties of the data. This happens typically at the level of the feature extraction layers commonly used in standard deep learning approaches. Specifically, we posit that in the case of APR and pose estimation, having a representation which is *equivariant* to the group of rigid motions, *i.e.,* rotations and translations in 3D, may be an effective way to boost the network performance. This should play the role of an implicit data augmentation by means of group equivariance, and in turn alleviate the need for an explicit data augmentation for training.

Indeed, recently, there has been a growing interest in designing more geometric models that are equivariant to groups of such transformations. These approaches leverage theoretical contributions from group theory, representation theory, harmonic analysis and fundamental deep learning [38, 98, 99, 41, 100, 101, 102]. More specifically, group-equivariant neural networks, or Group-equivariant CNNs (G-CNNs), are part of the broader and promising field of geometric deep learning [103], that aims to exploit any underlying geometric relationship that

can exist within the data. In particular, the special Euclidean groups in 2 and 3 dimensions, denoted as SE(2) and SE(3) and encompassing respective rigid motions, are of particular interest in 3D computer vision [41, 104].

Despite the conceptual advances they represent, to the best of our knowledge, the use of deep equivariant features in the APR context is still considerably unexplored. This chapter proposes, for the first time, to investigate and justify the use of deep equivariant features for solving APR see Figure (5.1).

Contributions. Our contributions are summarized below:

(1) A formulation of how an equivariant CNN induces representations of planar camera motions, lying in SE(2), directly into the feature space. Section (5.4.1)

(2) An intuitive explanation is provided as to how SE(2)-equivariant features can be leveraged to recover any camera motion lying in SE(3). Section (5.4.2)

(3) A lightweight equivariant pose regression model, referred to as *E-PoseNet*, is introduced. Section (5.5)

(4) Extensive experimental evaluation of E-PoseNet showing its competitive performance as compared to existing APR methods on standard datasets. Section (5.6)

chapter organization. An overview of state-of-the-art APR methods and current exploitations of deep equivariant features is given in Section (5.2). Section (5.3) presents the formal definition of equivariance along with the formulation of APR. The theoretical justification as to how SE(2)-equivariant features can explicitly encode planar camera motions is presented in Section (5.4), whereas the full pose regression pipeline is introduced in Section (5.5). An extensive experimental validation is given in Section (5.6) along with a discussion of limitations. Section (5.7) concludes the chapter.

5.2 Related Work

The goal of this chapter is to exploit the power of equivariant features in the context of APR. Therefore, we split this section into: (1) a review of the relevant literature on APR, and (2) an overview of recent deep equivariant feature extraction methods applied to computer vision

problems.

Absolute Pose Regression. Since the rise of deep learning and CNNs in the early 2010's, many works have explored the application of CNNs for APR. This began with the introduction of PoseNet by Kendal *et al.* [105], who used the GoogLeNet model [106] as a feature extraction backbone coupled with a regression head to estimate the translation and rotation vectors. Most of the subsequent improvements lie in changes in the feature extraction architecture [105, 107, 108], modified objective functions [109, 110, 111], and additional intermediate representations [112, 113].

In [89], Sattler *et al.* provide an in-depth analysis of existing works on APR [107, 114, 115, 116, 117]. In particular, they show that structure-based and image retrieval methods are more accurate than APR. Moreover, they demonstrate that APR algorithms do not explicitly leverage knowledge about projective geometry. Instead, they learn a mapping between image content and camera poses directly from the data, and in the form of a set of base poses such that all training samples can be expressed as a linear combination of these reference entities. Wang *et al.* [118] proposed an approach to integrate dense correspondence-based intermediate geometric representations within an end-to-end trainable pipeline. However, this method still relies on classical (non-equivariant) features, and thereby requires a significant amount of data for generalization. Furthermore, methods such as [112, 113, 118] propose to learn intermediate representations that are indirectly equivariant, such as segmentation masks, object detections, and depth or normal maps. However, this comes at the cost of parameter redundancy. This core observation suggests that directly learning equivariant features may be a valuable direction to improve the accuracy of pose estimation while reducing the number of model parameters.

Deep Equivariant Features. There is a rich history in computer vision on the design of hand-crafted equivariant features (*e.g.,* Scale-Invariant Feature Transform (SIFT) [25], Oriented filters [27], Steerable filters [26], Rotation-equivariant Fields of Experts (R-FoE) [28], Lie groups-based filters [29, 30]). In the deep learning literature, convolutional layers [55] have been proven to be equivariant to image shifting, while max-pooling layers are only invariant to small shifts of the input image [119].

Although convolutional layers are inherently equivariant to translation, there is a significant amount of spatial information regarding the inputs that is not encoded by CNNs in a precise fashion [31, 32]. More specifically, local and global poolings, if added to CNNs, render translation information unrecoverable, discarding the foregoing equivariance [56].
A recent investigation shows that many neurons in CNNs learn slightly transformed (*e.g.,* rotated) versions of the same basic feature [33]. These are especially common in early vision, *e.g.,* in curve detectors, high-low frequency detectors, and line detectors.

There have been attempts to extend the G-CNNs to wider groups of transformations. In [34, 35], Mallat *et al.* extended CNNs to be equivariant to SE(2) using scattering transform with predefined wavelets. In [36, 37], Bekkers *et al.* also extended CNNs to be equivariant to the SE(2) group via B-splines. In [38], Cohen *et al.* proposed group convolutions network equivariant to the p4m discrete group via 90° rotations and flips, where they demonstrated the effectiveness of group convolutions for classification task.
More recently, the use of equivariant features has been investigated for solving various computer vision tasks such as 3D point cloud analysis [100], aerial object detection [101] and 2D tracking [102]. In [120], Esteves *et al.* proposed to use projection and embedding from 2D images into a spherical CNN latent space to estimate the relative orientations of the object. Similarly, Zhang *et al.* proposed to use spherical CNN for learning camera pose estimation in omnidirectional localization [121]. However, to the best of our knowledge, equivariant features have not yet been explicitly leveraged in the context of APR for single 2D input image, which is the very focus of this chapter.

5.3 Preliminaries

This section provides the necessary mathematical background. First, we introduce the elements of group theory needed for understanding our work, then introduce the notions of invariant and equivariant features. Then, we present the general framework for APR, and finally show the added value of relying on equivariant features in this context.

Notation. The following notation will be adopted: vectors and column images are denoted

by boldface lowercase letters $\mathbf{x}$, matrices by uppercase letters X, scalars by italic letters x or X, functions as $\mathcal{X}$ and spaces as $\mathbb{X}$. The special orthogonal group, the Euclidean group, and the special Euclidean group, of dimension n, are denoted as $\mathrm{SO}(n)$, $\mathrm{E}(n)$ and $\mathrm{SE}(n)$, respectively.

5.3.1 Elements of Group Theory

In this section, we provide the basics of group theory needed to understand our work.

Groups A group $\mathfrak{G}$ is a set equipped with an operation that takes two elements from the group $\mathfrak{g}, \mathfrak{h} \in \mathfrak{G}$ and combines them to produce a third element $\mathfrak{gh}$. To define a group, the operation must satisfy the following properties:

- Closure: The group is closed under the operation, i.e. $\forall \mathfrak{g}, \mathfrak{h} \in \mathfrak{G}$, $\mathfrak{gh} \in \mathfrak{G}$.
- Associativity: $\forall \mathfrak{g}, \mathfrak{h}, \mathfrak{l} \in \mathfrak{G}$, $(\mathfrak{gh})\mathfrak{l} = \mathfrak{g}(\mathfrak{hl})$.
- Identity: there exists a unique $\mathfrak{e} \in \mathfrak{G}$ satisfying $\mathfrak{eg} = \mathfrak{ge} = \mathfrak{g}$, $\forall \mathfrak{g} \in \mathfrak{G}$.
- Inverse: For each $\mathfrak{g} \in \mathfrak{G}$ there exists a unique inverse $\mathfrak{g}^{-1} \in \mathfrak{G}$ such that $\mathfrak{g}^{-1}\mathfrak{g} = \mathfrak{g}\mathfrak{g}^{-1} = \mathfrak{e}$.

Group Representations

Group representations describe abstract groups in terms of bijective linear transformations (i.e. automorphisms) of vector spaces; in particular, they can be used to represent group elements as invertible matrices so that the group operation can be represented by matrix multiplication.

More precisely, a representation of a group $\mathfrak{G}$ on a vector space V over a field $\mathbb{K}$ is a group homomorphism from $\mathfrak{G}$ to GL(V), the general linear group on V. That is, a representation is a map

$$\rho : \mathfrak{G} \to GL(V)$$

such that

$$\forall \mathfrak{g}_1, \mathfrak{g}_2 \in \mathfrak{G}, \quad \rho(\mathfrak{g}_1\mathfrak{g}_2) = \rho(\mathfrak{g}_1)\rho(\mathfrak{g}_2).$$

Therefore, ρ is preserving the group structure while mapping $\mathfrak{G}$ to $GL(V)$.

5.3.2 Invariant and Equivariant Features.

Given an image $\mathbf{x}$, captured by a camera, an APR method $\mathcal{P}$ predicts the 6-Degrees-of-Freedom (6-DoF) pose, *i.e.,* position and orientation, of the camera with respect to its environment.

Let us denote by $\mathbb{I} \subset \mathbb{R}^m$ the linear space of vectorized m-dimensional images (or image regions), and by $\mathbb{F} \subset \mathbb{R}^n$ the latent space of features – with dimension n. Considering a CNN-based feature extraction function $\mathcal{F}$, we write:

$$\begin{aligned} \mathcal{F}: \ \mathbb{I} &\rightarrow \mathbb{F} \\ \mathbf{x} &\mapsto \mathcal{F}(\mathbf{x}). \end{aligned}$$

Given $\mathfrak{G}$, a generic group of transformations and $\mathfrak{g}$, an element of $\mathfrak{G}$, we denote by $\phi_{\mathfrak{g}}^{(\mathbb{I})}$ and $\phi_{\mathfrak{g}}^{(\mathbb{F})}$ the actions of $\mathfrak{g}$ into the image and feature spaces, respectively.

Definition 1 *$\mathcal{F}$ is invariant to $\mathfrak{G}$ if and only if*

$$\forall \mathfrak{g} \in \mathfrak{G}, \forall \mathbf{x} \in \mathbb{I}, \quad \mathcal{F}(\phi_{\mathfrak{g}}^{(\mathbb{I})}\mathbf{x}) = \mathcal{F}(\mathbf{x}). \tag{5.1}$$

Definition 2 *$\mathcal{F}$ is equivariant to $\mathfrak{G}$ if and only if*

$$\forall \mathfrak{g} \in \mathfrak{G}, \forall \mathbf{x} \in \mathbb{I}, \quad \mathcal{F}(\phi_{\mathfrak{g}}^{(\mathbb{I})}\mathbf{x}) = \phi_{\mathfrak{g}}^{(\mathbb{F})}\mathcal{F}(\mathbf{x}). \tag{5.2}$$

Note that invariance can be seen as a special case of equivariance where $\phi_{\mathfrak{g}}^{(\mathbb{F})} = \mathcal{I}$, the identity mapping, $\forall \mathfrak{g} \in \mathfrak{G}$.

Equivariant Features for APR. Sattler *et al.* proposed the following formulation for the pose function $\mathcal{P}$ [89]:

$$\mathcal{P}(\mathbf{x}) = \mathbf{b} + \mathsf{P}.\mathcal{E}(\mathcal{F}(\mathbf{x})), \tag{5.3}$$

where the feature extractor $\mathcal{F}$ is first applied to the image $\mathbf{x}$ followed by a non-linear embed-

ding of the features $\mathcal{E}$ lifting them to a higher-dimensional space. Then, a linear projection into the space of camera poses, represented by a matrix P, is applied. Finally, a bias term $\mathbf{b}$ is added.

As presented in Section (5.2), the work in [89] demonstrated that classical APR is more closely related to pose approximation via image retrieval than to accurate pose estimation leveraging the 3D structure, thus the accuracy gap.

Our hypothesis is that this is likely due the lack of geometric information carried by classical CNN features. Indeed, the perceptive power of classical CNNs can most often be considered as a statistical phenomenon, whereas pose estimation is a geometrical problem.

In this work, we thus propose to replace the classical convolutional layers of the feature extractor $\mathcal{F}$ by their group-equivariant counterparts [38], then to assess how this affects both the accuracy and data efficiency of the model.

Therefore, assuming that $\mathcal{F}$ is equivariant to $\mathfrak{G}$, *i.e.*, verifies **Definition** (2), and applying the transformation $\phi_{\mathfrak{g}}^{(\mathbb{I})}$ to the image $\mathbf{x}$, the pose regression function $\mathcal{P}$ in (5.3) becomes:

$$\mathcal{P}(\phi_{\mathfrak{g}}^{(\mathbb{I})}\mathbf{x}) = \mathbf{b} + \mathsf{P} \cdot \mathcal{E}\left(\phi_{\mathfrak{g}}^{(\mathbb{F})}\mathbf{v}\right), \tag{5.4}$$

where $\mathbf{v} = \mathcal{F}(\mathbf{x})$. This suggests that any action of $\mathfrak{G}$ on the image has a direct effect in the latent space and that, in particular, camera motion transformations of images, *i.e.* changes in camera pose, explicitly induce actions on the feature vector $\mathbf{v}$, and implicitly on the regressed pose.

Considering $\mathfrak{G}$ as SE(2) or SE(3), we posit that such equivariant features will help improve the performance of APR.

5.4 Pose from SE(2)-Equivariant Features

We consider a piecewise planar scene, where the scene planes are parallel to the image plane. We then consider camera motions that locally preserve the latter.

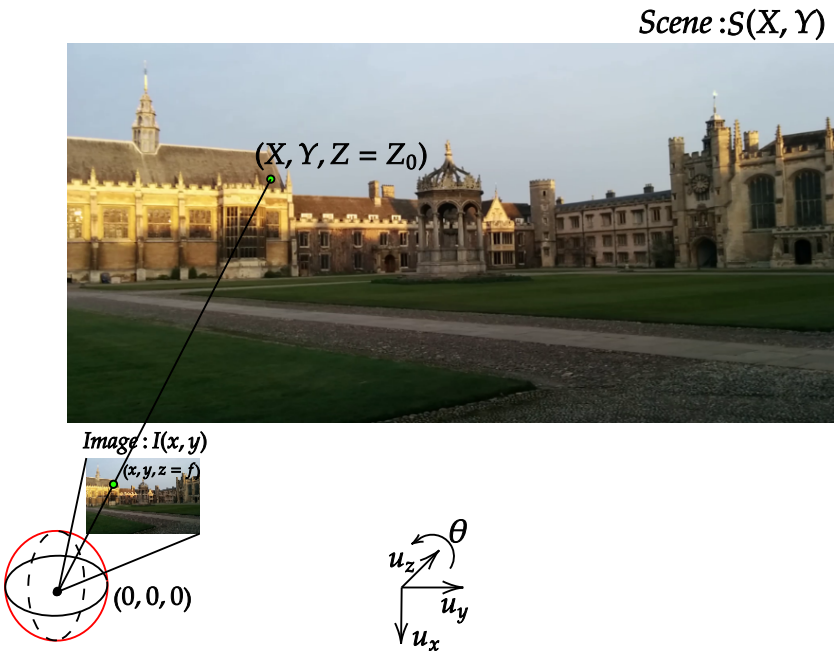


Figure 5.2: **Illustration** – Scene plane ($Z = Z_0$) - Parallel image plane ($Z = f$) - Camera center $(0, 0, 0)$. The scene is defined as the set of light intensity values $\mathsf{S}(X, Y)$ within the plane $Z = Z_0$. Rays of lights are then projected to the camera center. Intersections of projected rays with the image plane (considered as infinite) define image intensity values $\mathsf{I}(x, y)$. Camera motions are restricted to SE(2), *i.e.* translation along $\mathbf{u}_x$, $\mathbf{u}_y$ and rotation around $\mathbf{u}_z$ (characterized by roll angle θ).

5.4.1 SE(2)-Equivariant Features

We herein restrict camera motions to those of the SE(2) group, *i.e.,* planar translations and rotations within the image plane Figure (5.2). With that, we analyse the effects of planar camera motions on the image and feature spaces, assuming that the feature extractor $\mathcal{F}$ is equivariant to SE(2).

Effects of Camera Planar Motions on Images. Following the notations introduced in Figure (5.2), rotating the camera with a roll angle θ is equivalent to rotating the scene around $\mathbf{u}_z$ (camera viewing direction) with angle $-\theta$ [122].

Similarly, translating the camera center along $\mathbf{u}_x$ and $\mathbf{u}_y$ is equivalent to translating the scene

in the opposite direction.

Let us denote any rigid motion of the camera along its image plane *i.e.,* in SE(2) by $\mathsf{R}, \mathbf{t}$, where $\mathbf{t}$ is a planar translation and R a planar rotation. The effect of this motion on any point $\mathbf{p}$ of the scene is obtained by applying $-\mathbf{t}$ then $\mathsf{R}^\top$ such that $\mathbf{p}' = \mathsf{R}^\top(\mathbf{p} - \mathbf{t})$.

In 3D, considering $\mathbf{t} = (T_X, T_Y, 0)^\top$, we have

$$\mathbf{p}' = \begin{pmatrix} X' \\ Y' \\ Z' \end{pmatrix} = \begin{pmatrix} \cos(\theta) & \sin(\theta) & 0 \\ -\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X - T_X \\ Y - T_Y \\ Z \end{pmatrix}. \tag{5.5}$$

Following classical projection rules [123], image coordinates (x, y) are then given by $x = f\frac{X}{Z_0}$ and $y = f\frac{Y}{Z_0}$, where f is the distance from the camera center to the image plane, and Z_0 is the distance to the scene plane.

Multiplying (5.5) by $\frac{f}{Z_0}$, then restricting the coordinates to the first two ones gives:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{pmatrix} \begin{pmatrix} x - t_x \\ y - t_y \end{pmatrix}, \tag{5.6}$$

where $t_x = f\frac{T_X}{Z_0}$, and $t_y = f\frac{T_Y}{Z_0}$. By denoting $\mathsf{R}^{(2)}$ the 2D rotation matrix of angle θ and $\mathbf{t}^{(2)} = (t_x, t_y)^\top$, the effect of any planar camera motion $\mathsf{R}, \mathbf{t}$ on any point $\mathbf{p}^{(2)}$ of the image is thus given by $\mathbf{p}^{(2)\prime} = \mathsf{R}^{(2)\top}(\mathbf{p}^{(2)} - \mathbf{t}^{(2)})$, where $\mathbf{p}^{(2)\prime}$ is the image of $\mathbf{p}^{(2)}$ under the transformation.

We finally denote $\phi^{(\mathbb{I})}_{\mathsf{R},\mathbf{t}}$ the effect of the camera motion on an image $\mathbf{x}_1$, resulting in another image $\mathbf{x}_2$, *i.e.,* $\mathbf{x}_2 = \phi^{(\mathbb{I})}_{\mathsf{R},\mathbf{t}}\mathbf{x}_1$.

In what follows, we prove that the image transformation due to planar motions of the camera commute with the projection operator P in Figure (5.3). Indeed, applying a planar motion $\mathsf{R}_1, \mathbf{t}_1$ followed by a second one $\mathsf{R}_2, \mathbf{t}_2$ to the camera, has the following effect on any point $\mathbf{p}$ of the scene:

$$\mathbf{p}' = \mathsf{R}_1^\top (\mathbf{p} - \mathbf{t}_1),$$

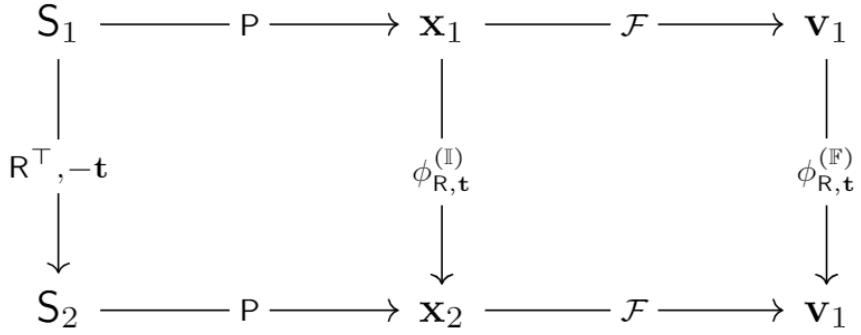


Figure 5.3: **Equivariance map** – Camera planar motion transformations of planar scenes ($\mathsf{S}_1 \mapsto \mathsf{S}_2$, first column), images ($\mathbf{x}_1 \mapsto \mathbf{x}_2$, second column) and features ($\mathbf{v}_1 \mapsto \mathbf{v}_2$, third column) induce representations of SE(2). In other words, these transformations commute with scene projector P and feature extractor $\mathcal{F}$.

then

$$\begin{aligned}
\mathbf{p}'' &= \mathsf{R}_2^\top \left(\mathbf{p}' - \mathbf{t}_2\right) \\
&= \mathsf{R}_2^\top \left(\mathsf{R}_1^\top \left(\mathbf{p} - \mathbf{t}_1\right) - \mathbf{t}_2\right) \\
&= \mathsf{R}_2^\top \left(\mathsf{R}_1^\top \mathbf{p} - \mathsf{R}_1^\top \mathbf{t}_1 - \mathbf{t}_2\right) \\
&= \mathsf{R}_2^\top \mathsf{R}_1^\top \left(\mathbf{p} - \left(\mathbf{t}_1 + \mathsf{R}_1 \mathbf{t}_2\right)\right) \\
&= \mathsf{R}_3^\top \left(\mathbf{p} - \mathbf{t}_3\right),
\end{aligned}$$

by denoting

$$\begin{cases}
\mathsf{R}_3 &= \mathsf{R}_2 \mathsf{R}_1 \\
\mathbf{t}_3 &= \mathbf{t}_1 + \mathsf{R}_1 \mathbf{t}_2.
\end{cases}$$

Indeed, 2D rotations commute, thus $\mathsf{R}_1 \mathsf{R}_2 = \mathsf{R}_2 \mathsf{R}_1$.

Similarly, one can easily observe that combining two camera motions has a similar effect on any point $\mathbf{p}^{(2)}$ of the image:

$$\mathbf{p}^{(2)\prime} = {\mathsf{R}_1^{(2)}}^\top \left(\mathbf{p}^{(2)} - \mathbf{t}_1^{(2)}\right),$$

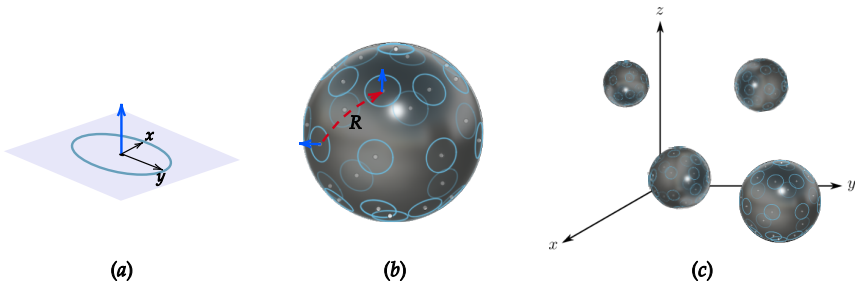


Figure 5.4: **From planar to 3D motions** – (Figure reproduced from [124]): (a) $\mathbb{S}(1)$, (b) $\mathrm{SO}(3)/\mathrm{SO}(2) \simeq \mathbb{S}(2)$, (c) $\mathrm{SE}(3) = \mathbb{R}^3 \rtimes \mathrm{SO}(3)$.

then

$$\begin{aligned}
\mathbf{p}^{(2)''} &= \mathsf{R}_2^{(2)\top}\left(\mathbf{p}^{(2)'} - \mathbf{t}_2^{(2)}\right) \\
&= \mathsf{R}_2^{(2)\top}\left(\mathsf{R}_1^{(2)\top}\left(\mathbf{p}^{(2)} - \mathbf{t}_1^{(2)}\right) - \mathbf{t}_2^{(2)}\right) \\
&= \mathsf{R}_2^{(2)\top}\left(\mathsf{R}_1^{(2)\top}\mathbf{p}^{(2)} - \mathsf{R}_1^{(2)\top}\mathbf{t}_1^{(2)} - \mathbf{t}_2^{(2)}\right) \\
&= \mathsf{R}_2^{(2)\top}\mathsf{R}_1^{(2)\top}\left(\mathbf{p}^{(2)} - \left(\mathbf{t}_1^{(2)} + \mathsf{R}_1^{(2)}\mathbf{t}_2^{(2)}\right)\right) \\
&= \mathsf{R}_3^{(2)\top}\left(\mathbf{p}^{(2)} - \mathbf{t}_3^{(2)}\right),
\end{aligned}$$

$$\begin{cases}
\mathsf{R}_3^{(2)} &= \mathsf{R}_2^{(2)}\mathsf{R}_1^{(2)} \\
\mathbf{t}_3^{(2)} &= \mathbf{t}_1^{(2)} + \mathsf{R}_1^{(2)}\mathbf{t}_2^{(2)}.
\end{cases}$$

Therefore,

$$\phi^{(\mathbb{I})}_{(\mathsf{R}_2,\mathbf{t}_2)\circ(\mathsf{R}_1,\mathbf{t}_1)} = \phi^{(\mathbb{I})}_{\mathsf{R}_3,\mathbf{t}_3} = \phi^{(\mathbb{I})}_{\mathsf{R}_2,\mathbf{t}_2} \circ \phi^{(\mathbb{I})}_{\mathsf{R}_1,\mathbf{t}_1}. \tag{5.7}$$

This proves that the correspondence from $\mathsf{R}, \mathbf{t}$ to $\phi^{(\mathbb{I})}_{\mathsf{R},\mathbf{t}}$ is a group homomorphism from SE(2). In other words, the set of $\phi^{(\mathbb{I})}_{\mathsf{R},\mathbf{t}}$, where $\mathsf{R}, \mathbf{t} \in \mathrm{SE}(2)$, is the image of a representation of SE(2) into the image space.

Effects of Camera Planar Motions on Features. We herein consider an SE(2)-equivariant CNN-based feature extractor $\mathcal{F}$. For the sake of clarity and simplicity, we discard the dis-

creteness of numerical images and consider their supports as continuous.

Classical convolutional layers are only equivariant to the translation group $(\mathbb{R}^2, +)$. Indeed, at each layer l, a conventional CNN takes as input a stack of intermediate feature maps $\mathbf{v}^{(l)} : \mathbb{R}^2 \to \mathbb{R}^{K^{(l)}}$ and convolves it with a set of $K^{(l+1)}$ filters $\psi^{(l)} : \mathbb{R}^2 \to \mathbb{R}^{K^{(l)}}$. Therefore we have

$$\forall \mathbf{t}^{(2)} \in \mathbb{R}^2, \quad \left((\phi_{\mathbf{t}^{(2)}} \mathbf{v}) * \psi^{(l)} \right)(.) = \left(\phi_{\mathbf{t}^{(2)}} \left(\mathbf{v} * \psi^{(l)} \right) \right)(.), \tag{5.8}$$

where $\phi_{\mathbf{t}^{(2)}}$ are images of $\mathbf{t}^{(2)}$ under representations of $(\mathbb{R}^2, +)$. In other words, if the input image is translated, the output feature map translates in the same way. However, in general, the same is not true for rotations, *i.e.* if the input image is rotated, the output feature map will not be rotated accordingly. The work in [41] has extended CNN equivariance to the SE(2) group, *i.e.* the group of continuous rotations and translations in $\mathbb{R}^2$, the image domain.

By replacing the translation group equivariance of classical CNNs by equivariance to SE(2), such particular CNNs can then be characterized by the following equation:

$$\begin{aligned} &\forall \mathsf{R}^{(2)}, \mathbf{t}^{(2)} \in \mathrm{SE}(2), \\ &\left(\left(\phi_{\mathsf{R}^{(2)}, \mathbf{t}^{(2)}} \mathbf{v} \right) * \psi^{(l)} \right)(.) = \left(\phi_{\mathsf{R}^{(2)}, \mathbf{t}^{(2)}} \left(\mathbf{v} * \psi^{(l)} \right) \right)(.), \end{aligned} \tag{5.9}$$

where $\phi_{\mathsf{R}^{(2)}, \mathbf{t}^{(2)}}$ are images of $\mathsf{R}^{(2)}, \mathbf{t}^{(2)}$ under representations of SE(2). In particular, considering the last convolutional layer output, we obtain that feature extraction $\mathcal{F}$ and Euclidean transformations of images commute.

Finally, the camera motion transformations of both images and features induce representations of SE(2). As a result, the image and feature spaces explicitly encode the planar motions of the camera.

5.4.2 From SE(2) to SE(3)

After demonstrating how an SE(2)-equivariant CNN can induce representations of planar camera motions directly into the feature space, we herein discuss how these features, which are equivariant to planar camera motions, *i.e.,* in SE(2), are leveraged for general pose regression in SE(3).

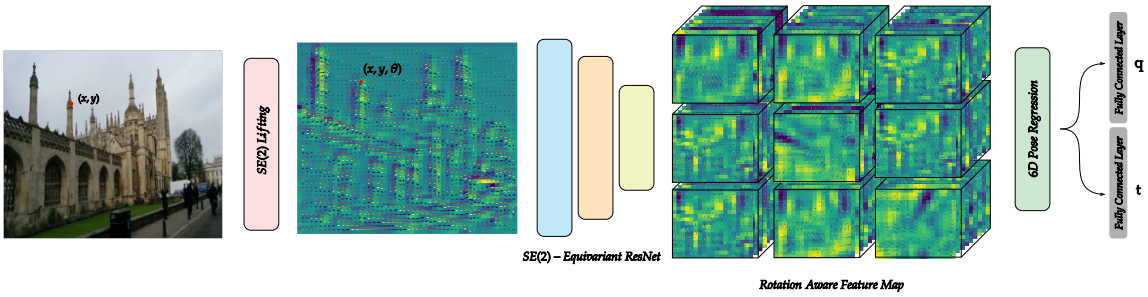


Figure 5.5: **E-PoseNet -** Our pose regression pipeline leverages a roto-translation equivariant ResNet18 [125] backbone, two fully connected Multilayer Perceptrons (MLP) for lifting the features to a higher dimensional space, followed by two branches for separately regressing the position and orientation of the camera.

Indeed, SE(2)-equivariant features extracted with $\mathcal{F}$, are now to be mapped to camera poses in SE(3) via $\gamma := \mathsf{P} \cdot \mathcal{E}$. This is the last step towards finding $\mathcal{P}(\mathbf{x})$, as defined in equation (5.3) [1].

The SE(3) group may be written as a semidirect product $\mathrm{SE}(3) = \mathbb{R}^3 \rtimes \mathrm{SO}(3)$, and similary for $\mathrm{SE}(2) = \mathbb{R}^2 \rtimes \mathrm{SO}(2)$. We may therefore restrict the discussion to the mapping $\gamma^* : \mathrm{SO}(2) \rightarrow \mathrm{SO}(3)$ [124, 103]. We rely on the observation that the quotient space $\mathrm{SO}(3)/\mathrm{SO}(2) \simeq \mathbb{S}(2)$ is the sphere in 3D, where $\simeq$ denotes a homeomorphism.

For every point on $\mathbb{S}(2)$, it is possible to move to another point via a rotation. $\mathbb{S}(2)$ is consequently a homogeneous space for SO(3), and SO(3) can be seen as a bundle of elements of $\mathbb{S}(1)$, *i.e.,* planar circles on $\mathbb{S}(2)$, for which a continuous mapping γ^* exists [124]. These mappings, γ^*, directly relate to γ by composing with translations. Figure (5.4) illustrates how the planar rotations around a fixed axis Figure (5.4)(a) can be viewed as local patches on the sphere in 3D Figure (5.4)(b); thus, relating to rotations in 3D, and finally how this may be generalized to full rigid motions by translation as shown in Figure (5.4)(c). The mapping γ is learned as part of the end-to-end APR.

Intuitively, one can interpret this as an approximation of the space of camera poses by a finite set of learned ones, with feature equivariance used to generalize and extend the coverage within the space. Indeed, an SE(2)-equivariant model is capable of generalizing

[1] For the sake of clarity, and without loss of generality, we drop the bias $\mathbf{b}$ in this discussion.

from each learned pose to every poses that preserve the image plane (*i.e.* z-rotated and x,y-translated versions of the original camera). In other words, instead of learning some cropped image planes like classical CNNs do, relying on an SE(2)-equivariant CNN rather consists in learning several infinite image planes, therefore providing a denser coverage of the scene space.

5.5 Proposed *E-PoseNet*

This section gives an overview of our proposed equivariant pose regression model, *E-PoseNet*. To be able to assess how explicitly encoding pose information into the feature space can result in a more accurate and data-efficient pose regressor, we proceed from the architecture of PoseNet [105]. We follow the same pipeline, except that we substitute the GoogLeNet backbone by an SE(2)-equivariant [41] version of ResNet [125], to extract both translation and rotation-equivariant features. The resulting model is presented in Figure (5.5).

Network Architecture. *E-PoseNet* is composed of a roto-translation equivariant ResNet18 backbone, two fully connected Multilayer Perceptrons (MLP) for lifting the features to a higher dimensional space, followed by two branches for separately regressing the position and orientation of the camera. Each branch consists of an independent fully-connected MLP head.

SE(2)-Equivariant Backbone. Our backbone, *i.e.,* feature extractor $\mathcal{F}$, is an SE(2) roto-translation equivariant version of ResNet. Specifically, we use the *e2cnn* [41] implementation for E(2)-equivariant convolution, pooling, normalization, and non linearities, to build an equivariant ResNet18.

To decrease the computational cost, we discretize the SE(2) group making the model only equivariant to the $(\mathbb{R}^2, +) \rtimes C_N$ group, meaning all translations in $\mathbb{R}^2$ and rotations by angles multiple of $\frac{2\pi}{N}$. Extracted features are now rotation-equivariant feature maps V with the size $(K \times N \times H \times W)$, where K is the number of channels, N the number of feature orientations (for our model we used $N = 8$), and H, W respectively, the height and width.

In addition to classical translation information, obtained features thus encode rotation infor-

mation that can enhance the pose regression. Furthermore, equivariance to broader transformations constrains the network in a way that can aid generalization, especially due to the weights shared under image rotations [38]. Finally, this rotation-equivariant ResNet shows a significant reduction in model size, about $1/N$ parameters compared to the regular ResNet architecture, to obtain the same feature size. Indeed, the size of classical feature maps is in the form $(K \times H \times W)$.

Loss Function. To regress camera poses, we use the loss function introduced in [115], and defined as:

$$\mathcal{L}_{\mathcal{P}} = \mathcal{L}_{\mathbf{t}} \exp\left(-s_{\mathbf{t}}\right) + s_{\mathbf{t}} + \mathcal{L}_{\mathsf{R}} \exp\left(-s_{\mathsf{R}}\right) + s_{\mathsf{R}}, \tag{5.10}$$

where the position loss $\mathcal{L}_{\mathbf{t}} = \|\mathbf{t}_0 - \mathbf{t}\|_2$, and the orientation loss $\mathcal{L}_{\mathsf{R}} = \left\|\mathbf{q}_0 - \frac{\mathbf{q}}{\|\mathbf{q}\|_2}\right\|_2$, are computed from predicted ($\mathbf{q}$,$\mathbf{t}$) and groundtruth ($\mathbf{q}_0$,$\mathbf{t}_0$) camera poses, considering the quaternion representation for orientations. $s_{\mathbf{t}}$, s_{R} are learned parameters.

5.6 Experiments and Analysis

The proposed method aims to improve APR accuracy by utilizing an equivariant feature extraction backbone able to learn geometry-aware feature maps. We first show the effect of SE(2)-equivariant models on rotated feature maps using samples from the T-Less dataset [126]. Then, we benchmark our proposed *E-PoseNet* on two datasets for both indoor and outdoor camera localization.

Equivariance analysis on T-Less. In this study, we use a sequence of 'object 5' from the T-less training dataset [126]. With only one textureless symmetric object present in the scene and undergoing continuous rotations, this sequence represents an ideal case for testing the impact of the different rotation parametrization and channeling. To assess the effect of equivariance, our backbone is made of 10 convolution layers with kernel size equal to 2, ELU non-linearity and Max Pooling downsampling every two layers with kernel size equal to 2. Different degrees of equivariance were tested, namely, Classical CNN "translation equivariant", Equivariant 90° (N=4), Equivariant 45° (N=8), Equivariant 18° (N=20), Equivariant 10° (N=36), and finally the Equivariant SO(2). Equivariant models are generated based

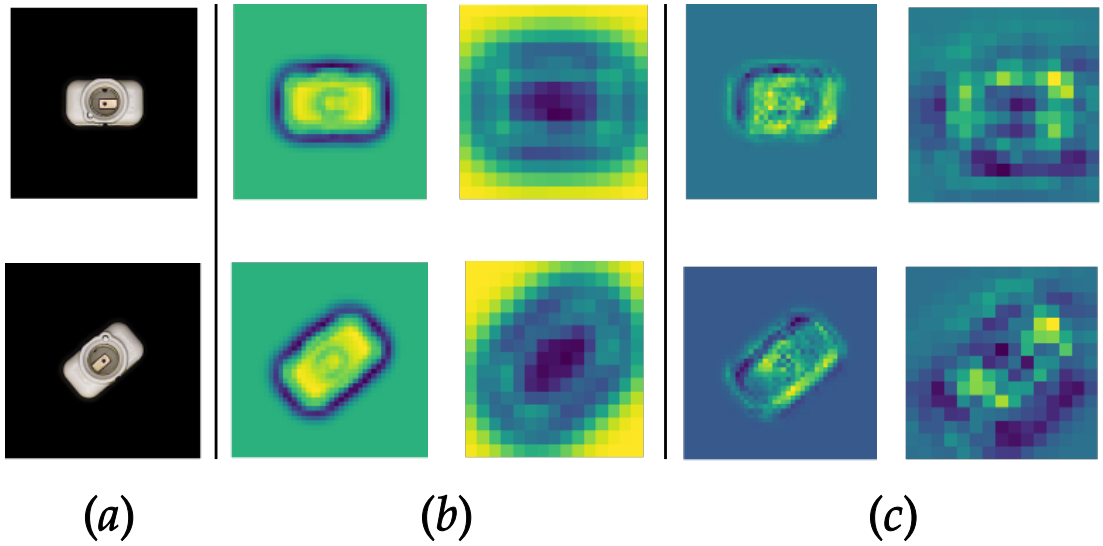


Figure 5.6: **Extracted feature maps -** difference between: (b) equivariant CNN and (c) classical CNN. The used samples (a) are from the T-LESS Dataset [126].

on *e2cnn* implementation [41]. We trained the model on one sequence only, without any data augmentation and for 100 epochs. Number of parameters, optimizer, learning rate and random seeds were fixed.

The difference between rotation-equivariant and classical CNN features is highlighted in Figure (5.6). By using images with different orientations Figure (5.6)(a) as input, the same transformation links extracted feature maps from different stages of the model (b). In contrast, this is not the case with the classical CNN where the obtained feature maps are not rotated versions of each other (c).

Figure (5.7) reports the proportion of samples for which the predicted pose error is below 10cm, 10°. We observed that increasing the level of equivariance, *i.e.,* decreasing the discrete sampling angle, leads to increasing the performance of the pose estimation model. Furthermore, the best reported performance has been achieved by the SO(2) continuous rotation equivariance. The metric used here does not follow the standard T-LESS metric since it is only used for model variants comparison.

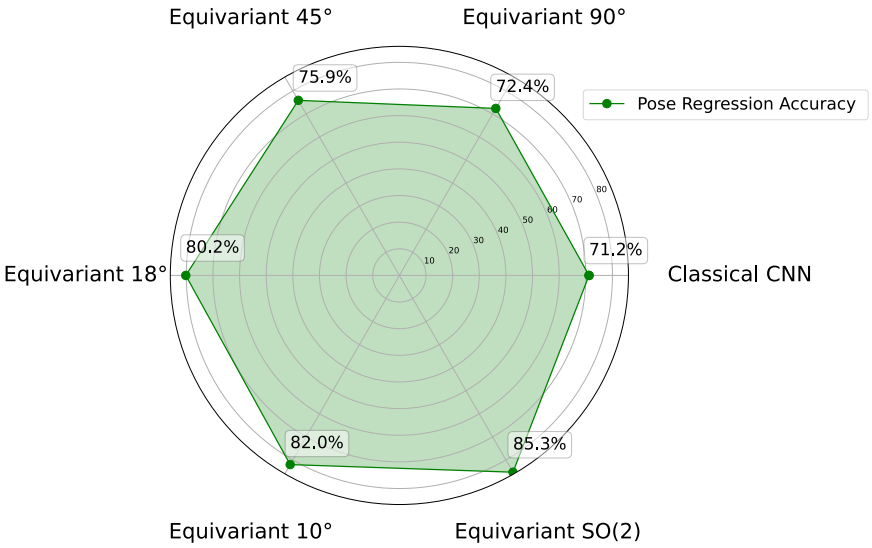


Figure 5.7: **Equivariant models comparison -** On sequence of 'object 5' from the T-less train dataset [126], we can see the increase in the model accuracy when increasing the equivariant group.

Datasets.

Cambridge Landmarks – We use this dataset [105] to evaluate the performance of *E-PoseNet* in outdoor camera relocalization. It is a large scale dataset taken around Cambridge University, containing original videos labelled with 6-DoF camera poses and a visual reconstruction of the scene (spatial extent of $\sim 900 - 5500m^2$). We train and evaluate *E-PoseNet* on four scenes (see Table (5.1)). Furthermore, a few samples are used to visualize the obtained *E-PoseNet* feature fields. Figure (5.8) shows that they directly support representations of SE(2) and are therefore enriched with some notion of orientation, visualized in a vector field form. On the contrary classical CNNs do not encode geometric information directly into their feature space.

7-Scenes – For indoor camera localization, we use the 7-Scenes dataset [136] which is a collection of tracked RGB-D camera frames for indoor scenes with a spatial extent of $\sim 1 - 10m^2$. Only RGB images are used in our experiments.

	King's College	*Old Hospital*	*Shop Facade*	*St. Mary*
DenseVLAD + Inter. (Baseline) [127]	1.48/4.45	2.68/4.63	0.90/4.32	1.62/6.06
PoseNet (PN) [105]	1.92/5.40	2.31/5.38	1.46/8.08	2.65/8.48
PN learned weights [115]	0.99/**1.06**	2.17/2.94	1.05/3.97	1.49/3.43
BayesianPN [128]	1.74/4.06	2.57/5.14	1.25/7.54	2.11/8.38
LSTM-PN [129]	0.99/3.65	1.51/4.29	1.18/7.44	1.52/6.68
SVS-Pose [130]	1.06/2.81	1.50/4.03	0.63/5.73	2.11/8.11
GPoseNet [131]	1.61/2.29	2.62/3.89	1.14/5.73	2.93/6.46
MapNet [108]	1.07/1.89	1.94/3.91	1.49/4.22	2.00/4.53
IRPNet [132]	1.18/2.19	1.87/3.38	0.72/3.47	1.87/4.94
MS-Transformer [133]	0.83/1.47	1.81/**2.39**	0.86/3.07	1.62/3.99
TransPoseNet [134]	**0.60**/2.43	1.45/3.08	**0.55**/3.49	1.09/4.99
E-PoseNet (Ours)	0.95/1.63	**1.43**/2.64	0.60/**2.78**	**1.00**/**3.16**

Table 5.1: **Comparative analysis of pose regressors on Cambridge Landmarks dataset (outdoor localization) [105] -** We report the median position/orientation error in meters/degrees for each method. Best results are highlighted in bold.

	Chess	*Fire*	*Heads*	*Office*	*Pumpkin*	*Kitchen*	*Stairs*
DenseVLAD + Inter. [127]	0.18/10.0	0.33/12.4	0.15/14.3	0.25/10.1	0.26/9.42	0.27/11.1	**0.24**/14.7
PoseNet (PN) [105]	0.32/8.12	0.47/14.4	0.29/12.0	0.48/7.68	0.47/8.42	0.59/8.64	0.47/13.8
PN learned weights [115]	0.14/4.50	0.27/11.8	0.18/12.1	0.20/5.77	0.25/4.82	0.24/5.52	0.37/10.6
BayesianPN [128]	0.37/7.24	0.43/13.7	0.31/12.0	0.48/8.04	0.61/7.08	0.58/7.54	0.48/13.1
LSTM-PN [129]	0.24/5.77	0.34/11.9	0.21/13.7	0.30/8.08	0.33/7.0	0.37/8.83	0.40/13.7
GPoseNet [131]	0.20/7.11	0.38/12.3	0.21/13.8	0.28/8.83	0.37/6.94	0.35/8.15	0.37/12.5
GeoPoseNet [115]	0.13/4.48	0.27/11.3	0.17/13.0	0.19/5.55	0.26/4.75	0.23/5.35	0.35/12.4
MapNet [108]	**0.08**/3.25	0.27/11.7	0.18/13.3	0.17/**5.15**	0.22/4.02	0.23/**4.93**	0.30/12.1
IRPNet [132]	0.13/5.64	0.25/9.67	0.15/13.1	0.24/6.33	0.22/5.78	0.30/7.29	0.34/11.6
AttLoc [135]	0.10/4.07	0.25/11.4	0.16/11.8	0.17/5.34	0.21/4.37	0.23/5.42	0.26/10.5
MS-Transformer [133]	0.11/4.66	0.24/**9.60**	0.14/12.2	0.17/5.66	0.18/4.44	**0.17**/5.94	0.26/8.45
TransPoseNet [134]	**0.08**/5.68	0.24/10.6	**0.13**/12.7	0.17/6.34	0.17/5.60	0.19/6.75	0.30/**7.02**
E-PoseNet (Ours)	**0.08**/**2.57**	**0.21**/11.0	0.16/**10.3**	**0.15**/6.80	**0.16**/**3.82**	0.20/6.81	**0.24**/9.92

Table 5.2: **Comparative analysis of pose regressors on the 7-Scenes dataset (indoor localization) [136] -** We report the median position/orientation error in meters/degrees for each method. Best results are highlighted in bold.

Finally, the two datasets present various challenges, *i.e.,* occlusion, reflections, motion blur, lighting conditions, repetitive textures, and variations in viewpoint and trajectory.

Comparative Analysis of Camera Pose Regressors. We compare the performance of *E-PoseNet* with state-of-the-art APR methods for camera localization in both outdoor and indoor scenes. First, we tested the performance on the Cambridge Landmarks dataset, for which we provide the median position and orientation errors in Table (5.1). We also compare the performance of *E-PoseNet* with respect to the state-of-the-art monocular pose regressors reporting on the 7-Scenes dataset. Table (5.2) contains the results. From the results on both datasets, and in comparison with APR methods, we conclude that the proposed *E-PoseNet* achieves the lowest location error across all the outdoor and indoor scenes, and

Figure 5.8: **Feature visualization for Cambridge Landmarks -** Visualization of samples images from the Cambridge Landmarks dataset [105] (left), along with their respective SE(2) group representations learned by *E-PoseNet* (right).

the lowest orientation error across the majority of them. It also competes with most recent transformer-based architectures [133, 134] on these datasets.

Implementation Details. We tested different architectures for the equivariant backbone, with ResNet18 being the most suitable model in our experiments, from both model size and performance perspectives. We trained our model for $5 - 10$k epochs using Adam optimizer [137], with $\beta_1 = 0.9, \beta_2 = 0.999$, $\epsilon = 10^{-5}$ and a batch size of 256. During the training phase, we rescaled the image so that its smaller length is 256 pixels followed by a random 224×224 crop. No further data augmentation was used.

Limitations. While we focus on introducing equivariant operations for the feature extraction part of the APR pipeline, the following stages (*i.e.* embedding, regression) do not have the same property, resulting in breaking the equivariance of the overall pipeline. Another limitation of the proposed APR model is the longer time required for equivariant CNN models as compared to classical CNN ones. Note that this is only during training, while the inference time is similar for both types of models.

5.7 Conclusions

This chapter presents a new direction for the problem of camera pose regression leveraging equivariant features to encode more geometric information about the input image. By using an SE(2)-equivariant feature extractor, our model is able to outperform existing methods on both outdoor and indoor benchmarks. Furthermore, we conclude that the equivariant properties of deep learning models that are used for geometric reasoning offer a promising direction for reaching the potential of absolute pose regression.

Chapter 6

SEPNet: Spacecraft Equivariant Pose Estimation Network

In this chapter, we demonstrate the application of our novel Equivariant-Pose Net, introduced in Chapter (5), to the specific problem of spacecraft pose estimation. This application is crucial within the realms of Space Situational Awareness and on-orbit servicing, where precise pose estimation is a fundamental requirement.

Spacecraft pose estimation is a critical challenge in orbital missions, imperative for the success of docking operations and maneuvering activities. In this chapter, we introduce the Spacecraft Equivariant PoseNet (SEPNet), a deep learning model specifically tailored for enhancing spacecraft pose estimation. SEPNet is designed to address the shortcomings of traditional end-to-end methods and classical CNNs, which often struggle to accurately process the geometric complexities inherent in pose estimation tasks.

SEPNet distinguishes itself by focusing on translation and rotation equivariance, thereby directly incorporating camera motion representations into its feature space. This approach facilitates more efficient use of data and obviates the necessity for cumbersome intermediate representations often required in other models. Through rigorous validation against standard datasets, SEPNet demonstrates not only improved accuracy in pose estimation but also more efficient architectural framework. Its performance in accurately determining spacecraft poses,

particularly in challenging space environments, underscores SEPNet's potential as a pivotal tool in advancing the capabilities of orbital missions and on-orbit services.

6.1 Introduction

Traditional pose estimation techniques for spacecraft, grounded in 3D geometric principles, determine the spacecraft's orientation and position. With the advent of Deep Learning (DL), Direct Pose Regression (DPR) emerged, offering advantages such as simplicity, reduced computational overhead, and end-to-end learning, wherein the entire neural network directly maps the input to the 6-DoF pose. This end-to-end approach harnesses the power of DL to encapsulate complex relationships inherent in space dynamics.

However, while DPR methods can provide faster inference crucial for real-time orbital adjustments and are less prone to overfitting on intermediate tasks, they sometimes may not match the accuracy levels of traditional 3D geometric methods [89]. Multi-stage methods, in contrast, can accumulate errors from one stage to another, where an error at an early stage might escalate in subsequent stages. Yet, DPR eliminates the potential for such accumulative errors by aiming to predict the pose in a single step.

Given the increasing importance of specialized orbital missions like OOS and ADR, the accuracy in pose estimation is paramount. Minor inaccuracies could culminate in mission failures, endangering existing space assets. Recognizing these challenges, this chapter endeavors to bridge the gap, addressing the inherent limitations of current DPR methodologies, while also aim for lightweight models apt for resource-constrained space missions.

In Chapter (5), we presented the emerging significance of incorporating equivariance in neural network architectures, particularly for DPR.

This technique allowed for encoding richer geometric information about the input image. The underlying premise that using equivariant features in DL models can substantially enhance geometric reasoning has exciting implications for 6-DoF object pose estimation.

Despite the promising advancements in the field, applying deep equivariant features in the context of spacecraft pose estimation remains largely uncharted territory.

In this light, this chapter embarks on a first-of-its-kind exploration to investigate and justify the use of deep equivariant features for spacecraft pose estimation. Our work aims to bridge this gap, offering a new lens through which to view and address the challenges of accurate and efficient pose estimation, especially in resource-constrained environments.

Contributions. Our key contributions can be summarized as follows:

(1) We propose lightweight equivariant G-CNN for spacecraft pose estimation.

(2) We offer an explanation for the advantages of using equivariant features and models in space missions.

(3) We present an extensive experimental evaluation of our model, comparing its performance against existing methods using standard datasets for spacecraft pose estimation.

Chapter Organization. The remainder of this chapter is organized as follows:

In Section (6.2), we provide an overview of current state-of-the-art methods and the role of DL based methods in spacecraft pose estimation. Section (6.3) presents the formal definitions and mathematical formulations of equivariance and DPR within the context of spacecraft pose estimation. Extensive experimental validation of our model is discussed in Section (6.5), which also includes a discussion of its limitations. Finally, Section (6.6) offers conclusions and directions for future research.

6.2 Related Work

The widespread adoption of CNNs in the domain of spacecraft relative pose estimation has been particularly noticeable in recent years. This trend is largely attributed to notable challenges like the Satellite Pose Estimation Challenge (SPEC) [138, 139], organized by Stanford University's Space Rendezvous Laboratory (SLAB) and the European Space Agency's Advanced Concepts Team (ACT), as well as the SPARK Challenge described in Chapter (3.4). In these competitions, most participants opted for DL based methods. The SPEC, for instance, used the Spacecraft Pose Estimation Dataset (SPEED) [140, 141] to benchmark the performance of various algorithms.

Algorithms for spacecraft pose estimation utilizing DL can generally be classified into two major categories: (1) Hybrid Modular Approaches and (2) Direct Pose Regression Approaches. Hybrid Modular Approaches meld classical computer vision techniques with multiple deep-learning models to estimate pose. Conversely, Direct Pose Regression Methods rely solely on a singular DL model, trained in an end-to-end manner, for estimating the pose [23].

6.2.1 Hybrid Modular Approaches

Hybrid Modular Approaches for spacecraft pose estimation combine DL models with conventional computer vision techniques. Typically, these methods progress through three core stages:

(1) Spacecraft Localization: The apparent size of the spacecraft in the image can fluctuate considerably based on the relative distance between the chaser and target spacecraft. Such scaling variations can adversely influence pose estimation accuracy. To adress this, spacecraft localization employs DL powered object detection frameworks to demarcate the spacecraft within the image. For instance, Chen *et al.*. [142, 20] utilized Faster-RCNN [143] with an HRNet-W18-C [144] backbone for this purpose, while Gerard *et al.*. [145] and Park *et al.*. [146] employed YOLOv3 [147] with respective MobileNetV2 [148] and DarkNet-53 [147] backbones. Also in Gaudillière *et al.*. [149] we proposed regressing the parameters of a 3D-aware Gaussian implicit occupancy function in an entirely differentiable fashion.

(2) Keypoint Prediction: In this stage, DL models predict the 2D projections of a predefined set of 3D keypoints within the localized regions of the spacecraft. These keypoints either stem from the spacecraft's CAD model or are reconstructed using techniques like multi-view triangulation or Structure from Motion (SfM). Chen *et al.*. [142] harnessed Pose-HRNet-W32 for keypoint prediction, whereas Wang *et al.*. [150] leveraged a transformer-based keypoint-set predictor with a ResNet50 [151] backbone. Alternatively, Gerard *et al.* [145] adopted YOLOv3 with a segmentation and regression decoder branch.

(3) Pose Computation: This conclusive stage calculates the spacecraft's pose from the predicted 2D keypoints and their corresponding 3D landmarks. Established algorithms like RANSAC [152] are frequently used to discern and discard outliers. For the actual pose

computation, techniques such as IterativePnP and EPnP [153] are prevalent. Some recent innovations, like the work of Chen *et al.* [142], incorporated PnP with RANSAC, which is further refined with a geometric loss optimized using the SA-LMPE optimizer. In contrast, Wang *et al.*. [150] used EPnP [153] with RANSAC, followed by a Levenberg–Marquardt [154] refinement step.

6.2.2 Direct Pose Regression Approaches

The ascent of DL techniques, particularly CNNs, has led to a surge of interest in their application to DPR for spacecraft. The initial wave of this trend was catalyzed by the work of Kendall *et al.*. [105], who employed the GoogLeNet architecture [106] for feature extraction and added a regression layer to estimate the 6-DoF pose of spacecraft. Subsequent advancements have primarily focused on refining the feature extraction architectures [105, 108], tailoring objective functions [109, 110, 111], and incorporating intermediate geometric representations [112, 113].

In the realm of spacecraft pose estimation, direct methods that utilize a single DL model for end-to-end pose regression have become increasingly significant. These approaches eliminate the need for intermediate computational steps, such as object detection or keypoint localization, and they also alleviate the necessity for additional external data like camera calibration parameters or 3D models of the spacecraft.

For instance, a study by Phisannupawong *et al.* [155] deployed a GoogLeNet-based architecture to directly regress a 7D pose vector. Their research indicated that employing a weighted Euclidean-based loss function during the training phase was more effective than using an exponential loss function, especially for orientation estimation.

Taking an alternative route, Sharma *et al.*. [80] introduced an innovative approach that involved discretizing the pose space into specific classes. They applied an AlexNet-based convolutional neural network to classify images of spacecraft into these discretized pose labels. However, this method had its limitations in terms of scalability and the requirement for further pose refinement.

To address these challenges, Sharma *et al.* [156] later developed the Spacecraft Pose

Network (SPN), a comprehensive model featuring multiple sub-branches. Each sub-branch served a unique purpose, such as spacecraft localization, orientation classification, and even weight calculation for orientation classes. The model then synthesized these outputs to refine the final spacecraft pose.

Similarly, Proença *et al.* [62] proposed URSONet, which was built on a ResNet-based backbone and separated into two branches for the estimation of position and orientation. They also introduced a continuous orientation estimation method that employed soft-assignment coding, significantly improving the accuracy of orientation predictions. This was later optimized into a mobile-friendly variant known as Mobile-URSONet [157], which drastically reduced the model's size without sacrificing performance.

Recent advancements in the field include SPNv2 [141], a model that specifically tackles domain generalization issues by employing multi-task learning. The architecture features a shared feature extractor followed by multiple prediction heads for tasks like spacecraft presence classification, bounding box prediction, and pose estimation.

In a comprehensive study, Sattler *et al.* [89] highlight that the existing DPR methods approaches often fail to effectively exploit projective geometric principles. Instead, they predominantly rely on learning a direct mapping from images to ground truth poses, often expressed as a combination of a set of reference poses.

To improve the accuracy of DPR, this chapter focuses on harnessing the capabilities of SE(2)-equivariant CNN-based feature extractors.

6.3 Preliminaries

This section lays down the essential mathematical groundwork needed for this chapter.

We begin by defining the rigid body transformation between chaser camera C and target B, then move on to the general framework for DPR, and finally discuss the importance of equivariant features in this context.

Notation. The following notation will be adopted in the chapter: vectors and column images are denoted by boldface lowercase letters $\mathbf{x}$, matrices by uppercase letters $\mathbf{X}$, scalars by italic

letters x or X, functions as $\mathcal{X}$ and spaces as $\mathbb{X}$. The Special Orthogonal group, the Euclidean group, and the special Euclidean group, of dimension n, are denoted as SO(n), E(n) and SE(n), respectively.

The transformation from the target spacecraft frame B to the camera frame C can be represented as a rigid body transformation $\mathbf{T}_{BC}$ in SE(3):

$$\mathbf{T}_{BC} = \begin{bmatrix} \mathbf{R}_{BC} & \mathbf{t}_{BC} \\ \mathbf{0}^T & 1 \end{bmatrix}, \tag{6.1}$$

where $\mathbf{T}_{BC} \in$ SE(3).

- $\mathbf{R}_{BC} \in$ SO(3) is the rotation matrix that aligns the coordinate frame of C with that of B. In this chapter we will use unit quaternion $\mathbf{q}_{BC}$ for representing this rotation.

- $\mathbf{t}_{BC} \in \mathbb{R}^3$ is the translation vector that maps the origin of C to the origin of B.

The action of $\mathbf{T}_{BC}$ on a point $\mathbf{p}$ in frame C to transform it into frame B is given by:

$$\mathbf{p}_B = \mathbf{T}_{BC}\mathbf{p}_C = \mathbf{R}_{BC}\mathbf{p}_C + \mathbf{t}_{BC}. \tag{6.2}$$

Direct Pose Regression for Spacecraft Pose Estimation

In the context of estimating the pose of spacecraft, we follow the same formulation presented in Chapter (5), Section (5.3).

Here, the image $\mathbf{x}$, captured by the camera on the chaser spacecraft, is first processed by the feature extractor $\mathcal{F}$. The extracted features are then elevated to a higher-dimensional space through a nonlinear embedding function $\mathcal{E}$. This is followed by a linear projection, represented by matrix P, into the 6-DoF pose space. Finally, a bias term $\mathbf{b}$ is added to complete the pose estimation.

As highlighted in Section (6.2) the traditional DPR methods are often better suited for pose approximation via image retrieval rather than for accurate pose estimation that leverages the 3D geometric structure. This observation underscores the need for more advanced methods in spacecraft pose estimation to bridge this accuracy gap.

Invariant and Equivariant Features

In the context of a captured image $\mathbf{x}$ from a camera, a DPR system $\mathcal{P}$ estimates the spacecraft's 6-DoF pose, which encompasses both its position and orientation relative to the camera.

The central hypothesis in Chapter (5) is that the limited geometric information in conventional CNN features contributes to this gap. While classical CNNs predominantly operate on a statistical perception basis, pose estimation inherently involves geometrical estimation.

To address this, the proposal involves replacing the standard convolutional layers in the feature extractor $\mathcal{F}$ with group-equivariant layers. The impact of this change on both the accuracy and data efficiency of the model is then assessed.

With the assumption that $\mathcal{F}$ is equivariant to a group $\mathfrak{G}$ and conforms to (Definition 2), applying the transformation $\phi_{\mathfrak{g}}^{(\mathbb{I})}$ to the image $\mathbf{x}$ modifies the pose regression function $\mathcal{P}$ as:

$$\mathcal{P}(\phi_{\mathfrak{g}}^{(\mathbb{I})}\mathbf{x}) = \mathbf{b} + \mathsf{P} \cdot \mathcal{E}\left(\phi_{\mathfrak{g}}^{(\mathbb{F})}\mathbf{v}\right), \tag{6.3}$$

where $\mathbf{v} = \mathcal{F}(\mathbf{x})$.

This highlights a direct correlation between actions of $\mathfrak{G}$ on the image and resultant effects in the latent space. Specifically, transformations in camera motion (i.e., changes in camera pose) induce explicit actions on the feature vector $\mathbf{v}$, and implicitly on the regressed pose.

By considering groups like SE(2) or SE(3), is posited that these equivariant features could significantly enhance the performance of DPR systems.

6.4 Method

Our work employs an CNN-based feature extractor equivariant to group $\mathfrak{G}$, denoted as $\mathcal{F}_{\mathfrak{G}}$. The aim is to explore how encoding poses information explicitly into the feature space can improve the accuracy of spacecraft pose estimation, narrowing the gap in the field.

Our model follows the approach proposed in (5) for implementing and training our neural network models.

6.4.1 Motivation

The foundational principle behind G-CNNs is that of template matching of rotated versions of the same filter as illustrated in Figure (2.5), a process wherein a predefined kernel undergoes transformations as defined by a mathematical group and is then subjected to an inner-product operation under all possible group transformations [38].

This procedure culminates in generating higher-dimensional feature maps, essentially functions constructed over the transformation group. These elevated feature spaces could a subsequent, more advanced level of template matching, thereby facilitating the identification of complex geometric relationships among features at diverse relative poses.

Notably, G-CNNs are constructed using equivariant layers, which utilize weight sharing to ensure computational efficiency.

In addition, these networks incorporate pooling layers specifically designed to achieve invariant properties. Integrating these advanced computational techniques implies that G-CNNs hold considerable promise as a powerful tool for enhancing the accuracy and reliability of spacecraft pose estimation algorithms.

6.4.2 SEPNet for Spacecraft Pose Estimation.

SEPNet model, follow the same model architecture proposed in Chapter (5), adapted for spacecraft pose estimation. SEPNet backbone is an SE(2)-equivariant version of ResNet [125, 41] to capture both translational and rotational features that directly correlated to the spaceraft pose.

Loss Function. For regressing the pose of the spacecraft, we employ a loss function as defined in [115]:

$$\mathcal{L}_{\mathcal{P}} = \mathcal{L}_{\mathbf{t}} \exp(-s_{\mathbf{t}}) + s_{\mathbf{t}} + \mathcal{L}_{\mathsf{R}} \exp(-s_{\mathsf{R}}) + s_{\mathsf{R}}, \tag{6.4}$$

In the context of spacecraft pose estimation, the pose is characterized by a relative position

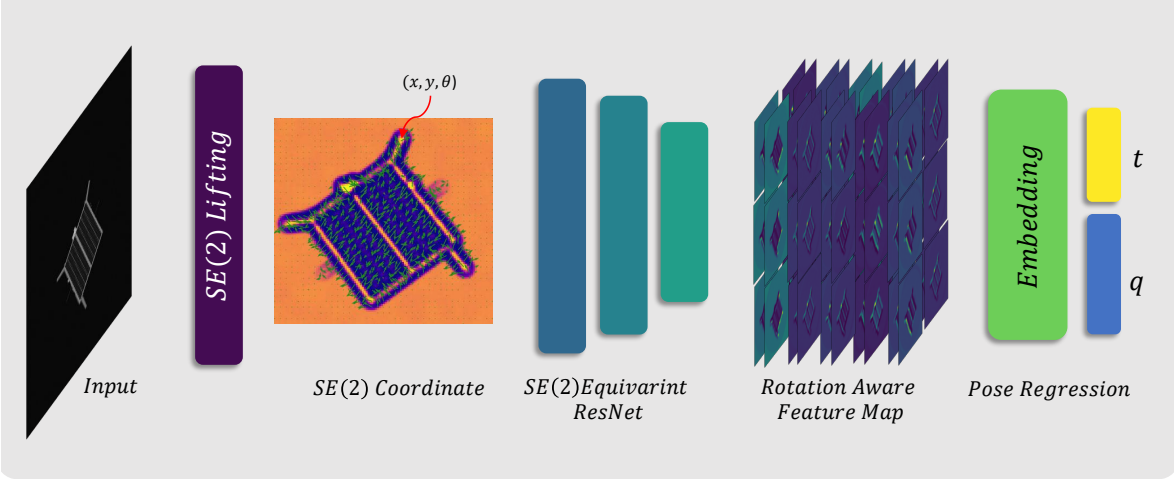


Figure 6.1: **Visual Representation of the Pose Regression Process**: Starting with an initial image input, the system employs SE(2) coordinate lifting to elevate the image to a relevant coordinate system. This is then passed through an SE(2)-equivariant ResNet backbone for feature extraction. Post this stage, the model learns a rotation-aware feature map to accentuate rotationally important aspects of the image. Finally, the system splits into two distinct branches for precise regression of both the camera's position and orientation, ensuring comprehensive camera pose determination.

vector $\mathbf{t}_{BC}$ and a unit quaternion $\mathbf{q}_{BC}$. Specifically, $\mathbf{t}_{BC}$ represents the coordinates of the origin of frame B in the frame C, while $\mathbf{q}_{BC}$ denotes the unit quaternion that defines the rotation necessary to align frame B with frame C.

The positional loss $\mathcal{L}_{\mathbf{t}} = \|\mathbf{t}_0 - \mathbf{t}\|_2$ and the orientational loss $\mathcal{L}_{\mathsf{R}} = \left\|\mathbf{q}_0 - \frac{\mathbf{q}}{\|\mathbf{q}\|_2}\right\|_2$ are computed based on the predicted $(\mathbf{q}, \mathbf{t})$ and ground truth $(\mathbf{q}_0, \mathbf{t}_0)$ poses of the spacecraft, utilizing quaternion representation for orientations. The terms $s_{\mathbf{t}}$ and s_{R} are trainable parameters.

6.5 Experiments and Analysis

The primary objective of our work is to study the potential benefits of explicitly encoding pose information into the feature space, thereby aiming to augment the accuracy of spacecraft pose estimation.

6.5.1 Datasets

The SPEED dataset [156] offers a comprehensive collection of high-resolution grayscale images (1920 × 1200 pixels) of the Tango spacecraft, comprising 15,000 synthetic and 300 real images. This dataset, developed in the advanced TRON facility, features a full-scale spacecraft mockup, sophisticated lighting simulations, and a seven-degree-of-freedom robotic camera arm, ensuring high accuracy in pose labels through calibration with Vicon camera systems. The SPEED dataset, while rich in synthetic data, underscores the challenges in acquiring real satellite imagery in orbit, as evidenced by its limited real-image collection. Specifically, the SPEED dataset is divided into training (12,000 synthetic, 5 real), validation (no specific split), and test (2,998 synthetic, 300 real) subsets.

Building on this, the SPEED+ [141] dataset marks a significant expansion in satellite pose estimation research. It not only adds 59,960 synthetic images but also introduces 6,740 Hardware-in-the-Loop (HIL) images, elevating the quality of illumination simulations and label accuracy. The SPEED+ dataset which serves as a basis for Satellite Pose Estimation Challenge, includes a training set with 47,966 synthetic images, a validation set with 11,994 synthetic images, and a test set comprising 6,740 synthetic images along with 2,791 images under Lightbox and Sunlamp conditions. This augmented dataset is a robust platform for developing and testing machine learning models, aimed at enhancing the precision of pose estimation techniques under diverse and challenging conditions.

6.5.2 Metrics

In the process of computing the pose score, two primary steps are involved. Initially, the scores related to orientation and position are calculated separately. The position error for a given image, denoted as i, is determined by the disparity between the estimated position vector and the actual (ground truth) position vector for each image from the dataset. This error is normalized against the actual distance of the satellite from the observer, expressed as:

$$\text{err}^{(i)}_{\text{position}} = \frac{\left| r^{(i)}_{gt} - r^{(i)}_{est} \right|_2}{\left| r^{(i)}_{gt} \right|_2}. \tag{6.5}$$

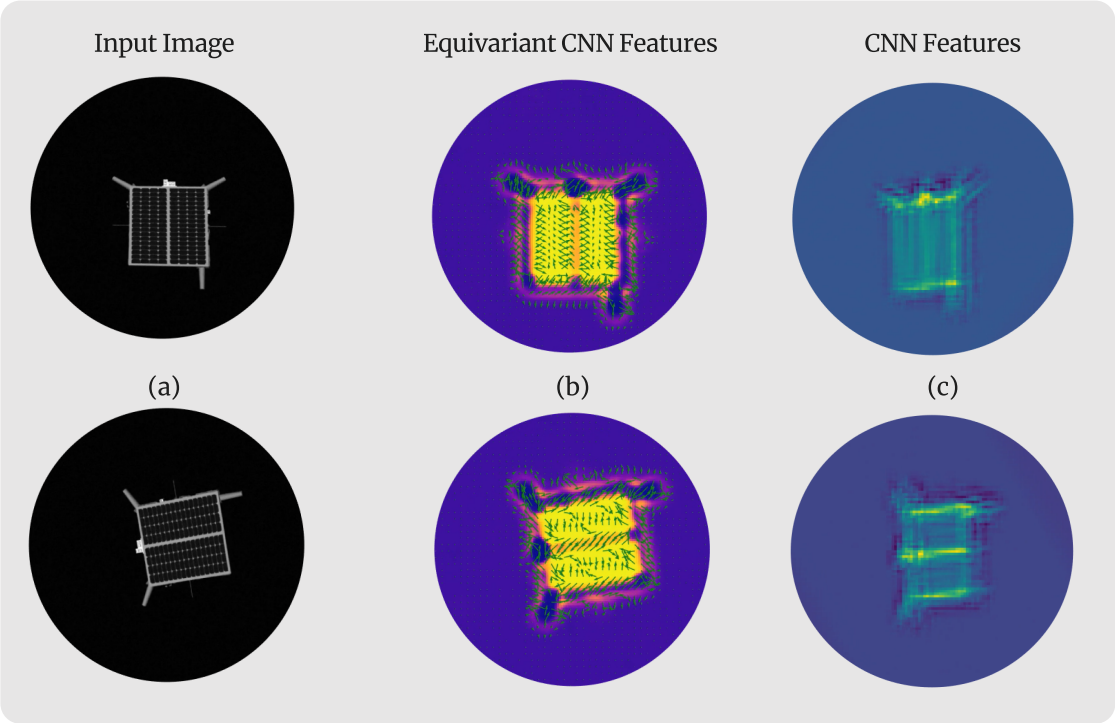


Figure 6.2: **Extracted Feature Maps:** Comparison between (b) Equivariant and (c) Classical CNN features. The test samples (a) is from the SPEED+ dataset.

The orientation error is quantified as the angular difference required to align the estimated and actual orientations. This is mathematically represented as:

$$\text{err}^{(i)}_{\text{orientation}} = 2 \cdot \arccos\left(\left|\left\langle q^{(i)}_{\text{est}}, q^{(i)}_{gt} \right\rangle\right|\right). \tag{6.6}$$

Consequently, the pose score for an image **i** is computed as the aggregate of its orientation and position scores:

$$\text{score}^{(i)}_{\text{pose}} = \text{score}^{(i)}_{\text{orientation}} + \text{score}^{(i)}_{\text{position}}. \tag{6.7}$$

Finally, the overall score is calculated as the mean of the pose scores across all images **N** in the test set:

$$\text{score} = \frac{1}{N}\sum_{i=1}^{N} \text{score}_{\text{pose}}^{(i)}\,. \tag{6.8}$$

6.5.3 Comparative Analysis of Spacecraft Pose Estimation Methods

In our experiments, we conducted a thorough assessment of our model's performance in spacecraft pose determination, comparing it against leading DPR techniques using the SPEED and SPEED+ datasets. We recorded the mean position and orientation errors, which are detailed in Table (6.1). Our analysis, which took into account both the datasets' characteristics and the various DPR, revealed that our model not only yields competitive results but also benefits from a smaller model size. Moreover, when juxtaposed with the latest models mentioned in recent surveys, such as in Pauly *et al.* [23], our model holds its ground, showcasing comparable, if not superior, performance.

This results underscores a substantial advancement in optimizing accuracy while concurrently mitigating computational resource demands. A particular result worthy of emphasis is the marked elevation in pose estimation accuracy observed, especially under challenging conditions, thus exemplifying the robustness inherent in our method.

A visual representation of the value of utilizing group equivariant CNN in Figure (6.2), elucidates the enhanced performance of our method in comparison to the baseline DPR method.

In our study, we also conducted a comparative analysis of different model architectures, namely equivariant ResNet models, named EResNet18, EResNet34, and EResNet50, focusing on how changes in model size affect performance. We observed a clear trend: as the model size increases, there is a corresponding improvement in model accuracy. This relationship between model size and accuracy is depicted in the accompanying Figure (6.3).

6.6 Conclusion

This chapter presents the Spacecraft Equivariant PoseNet (SEPNet) model, a deep learning approach for spacecraft pose estimation. SEPNet incorporates SE(2)-equivariant convolu-

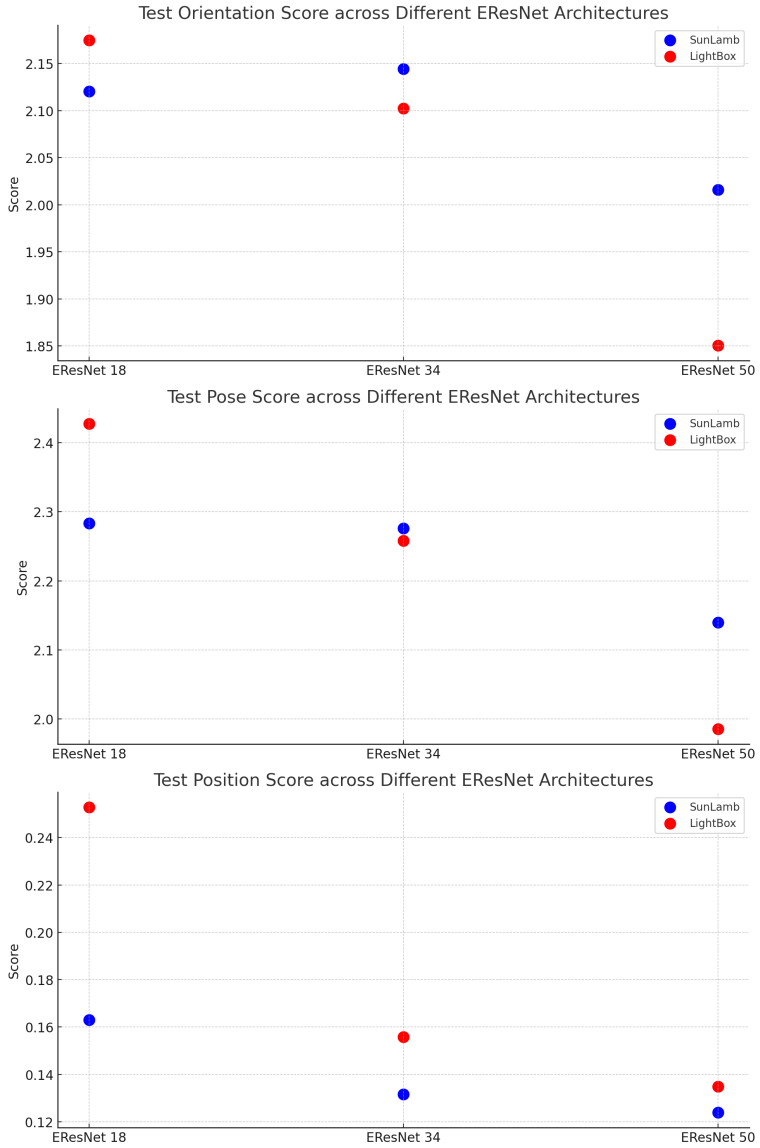


Figure 6.3: **Performance comparison of various EResNet architectures** (EResNet 18, EResNet 34, and EResNet 50) on two distinct data sources from SPEED+ dataset: Sunlamp and Lightbox. Each point represents the test score for orientation, pose, and position metrics. The results highlight the varying performance of the models across metrics and dataloaders, emphasizing the importance of architecture choice in relation to the specific dataset under consideration.

Reference	**Parameters (millions)**	**Mean position error ($E_{\mathbf{t}}$) (m)**	**Mean rotation error ($E_{\mathbf{R}}$) (deg)**
Sharma *et al.*. [a]	20.8	0.83 (Imitation-25 dataset)	14.35 (Imitation-25 dataset)
SPN	-NA-	0.7832	8.4254
SPNv2 [a]	52.5	0.031(SPEED+)	0.885 (SPEED+)
URSONet	$\sim$ 11.4 to $\sim$ 42.8($\sim$ 500[c])	0.1450[b]	2.4900[b]
Mobile-URSONet[a]	7.4	0.5600	6.2900
LSPnet	$\sim$ 47.8	0.4560	13.9600
Huang *et al.*.	$\sim$ 23.9	0.1715(URSO-OrViS dataset)	4.3820(URSO-OrViS dataset)
Phisannupawong *et al.*.	$\sim$ 7.0	1.1915[d](URSO-OrViS dataset)	13.7043[d](URSO-OrViS dataset)
SEPNet (Ours)	$\sim$ 14.0	0.1806 (SPEED) 0.0336 (SPEED+ Synthetic) 0.16304 (SPEED+ Sunlamp) 0.2528 (SPEED+ Lightbox)	2.3073 (SPEED) 1.8531 (SPEED+ Synthetic) 2.1204 (SPEED+ Sunlamp) 2.1746 (SPEED+ Lightbox)

Table 6.1: Comparison of Different Network Architectures for Spacecraft Pose Estimation and their performance

[a] Details of the best-performing variant reported.
[b] Results from KSPEC first edition .
[c] Number of parameters in the best performing ensemble of models reported by the authors.
[d] Median values reported.

tional neural networks, significantly improving pose estimation accuracy and computational efficiency in challenging space environments. The experimental results, utilizing datasets like SPEED and SPEED+, clearly demonstrate SEPNet's superiority over existing methods. This advancement holds great promise for enhancing the safety and reliability of critical space missions, including On-Orbit Servicing and Active Debris Removal. SEPNet represents an advancement in spacecraft navigation technology, with potential for widespread application in future space endeavors.

Chapter 7

Temporal Information for Trajectory Estimation of Space Objects

This chapter presents a new temporally consistent space object 3D trajectory estimation from a video taken by a single RGB camera. Understanding space objects' trajectories is an important component of Space Situational Awareness, especially for applications such as Active Debris Removal, On-orbit Servicing, and Orbital Maneuvers. Using only the information from a single image perspective gives temporally inconsistent 3D position estimation. Our approach operates in two subsequent stages. The first stage estimates the 2D location of the space object using a convolution neural network. In the next stage, the 2D locations are lifted to 3D space, using temporal convolution neural network that enforces the temporal coherence over the estimated 3D locations. Our results show that leveraging temporal information yields smooth and accurate 3D trajectory estimations for space objects. A dedicated large realistic synthetic dataset, named *SPARK-T*, containing 3 spacecrafts, under various sensing conditions, is also proposed and will be publicly shared with the research community.

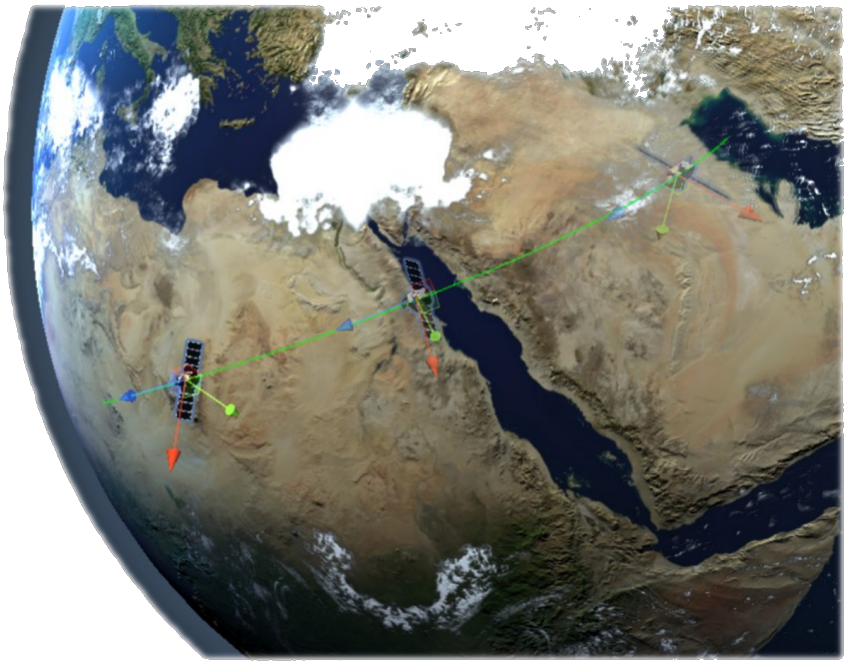


Figure 7.1: Spacecraft trajectory simulation

7.1 Introduction

Since the beginning of space exploration, the number of space debris has increased drastically. Debris population comes mainly from the remnants from human-made objects such as dead satellites, used rocket stages, and particles from the collision of other debris [158]. Today, these objects represent a threat as space debris incurs the risk of collision and damage to operational satellites.

To tackle this problem, one of the proposed solutions is Active Debris Removal (ADR). The premise of this method consists of capturing and disposing of large debris ($>$ 10cm). To that end, new technological challenges related to orbital rendezvous in general, and to relative navigation in particular, must be addressed. A reliable navigation system should be developed. It is required to be able to provide accurate relative state estimates of the targeted debris, over a wide range of different distances, from early detection until target capture.

Programs such as *CleanSpace* [159], *RemoveDebris* [52], *AnDROiD* [160], and future missions such as *ClearSpace-1* [161] lead the efforts to provide a cleaner space. Depending on the

specific mission objectives, debris state estimates can cover either the relative position and velocity (3-DoF relative navigation) of the targeted object, or the relative position, velocity, attitude (6-DoF relative navigation), as well as the target trajectory.

The contribution of this paper is twofold: First, we propose a new spatio-temporal approach for space object 3D trajectory estimation. Second, a large and, to the best of our knowledge, the first photo-realistic synthetic dataset with temporal information for space object 3D trajectory estimation was created and will be publicly shared with the research community. This dataset is named *SPARK-T*, where "T" stands for *trajectories*, presented in Chapter (3) Section (3.5).

In this Chapter, we focus on space object 3D trajectory estimation from videos where we exploit the temporal information. The proposed approach follows a top-down strategy. First, we start by detecting the center of a space object as 2D coordinates for each frame. Then, we lift the detected 2D coordinates to 3D space leveraging the temporal information contained in the observed video sequence. In order to test the proposed approach, a new dedicated dataset has been generated under a photo-realistic space simulation environment, with a large diversity in sensing conditions. Obtained experimental results show stable and accurate space object trajectory estimation. For ADR, such decomposition of the problem reduces the difficulty of the task at hand. It gives the possibility to control which estimation to use based on the orbital situation of the spacecraft.

7.2 Problem formulation

In this section, we formulate the considered problem of spacecraft 3D trajectory estimation. Let $\mathbf{V}_I = \{\mathbf{I}_1, \cdots, \mathbf{I}_N\}$ be a sequence of RGB images corresponding to the observed spacecraft or debris, where N is the total number of frames, and where the acquisition is done with a known camera whose intrinsic matrix is $K \in \mathbb{R}^{3\times4}$. Subsequently, the goal of this work is to estimate the trajectory of the object of interest in 3D. That is, the objective is to estimate the trajectory $Y = \{\ell_1, \cdots, \ell_N\}$, where $\ell_i \in \mathbb{R}^3$.

The object may be localized on each image I_i by estimating its pixel coordinates $(u_i, v_i) \in \mathbb{R}^2$.

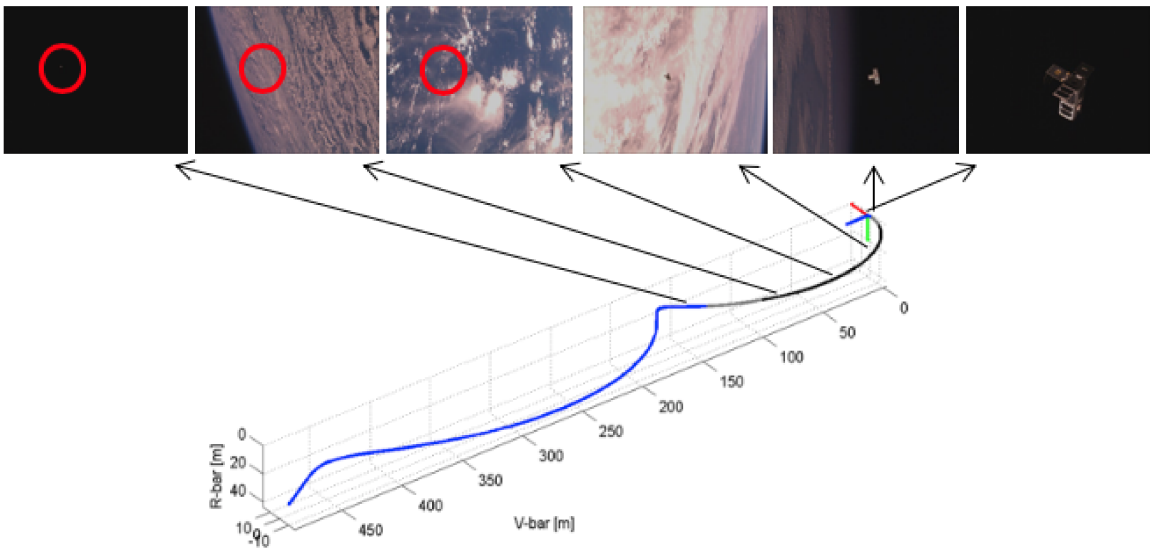


Figure 7.2: Tracking a spacecraft within vision-based navigation camera field of view over the reference trajectory from RemoveDebris mission [52].

This 2D location corresponds to the projection of the 3D location ℓ_i of the object in the scene onto a 2D image plane using the camera intrinsic parameters K such that

$$\begin{pmatrix} \grave{u}_i \\ \grave{v}_i \\ \grave{w}_i \end{pmatrix} = K(\mathbf{R}_i \ell_i + \mathbf{t}_i), \quad \text{and} \quad \begin{pmatrix} u_i \\ v_i \end{pmatrix} = \begin{pmatrix} \grave{u}_i / \grave{w}_i \\ \grave{v}_i / \grave{w}_i \end{pmatrix} \tag{7.1}$$

where $\mathbf{R}_i \in \mathrm{SO}(3)$ is the rotation matrix and $\mathbf{t}_i \in \mathbb{R}^3$ is the translation vector are the unknown space object rotation and translation at frame i, respectively, relative to the camera.

The task at hand can be formulated as a two-step problem: (1) Estimation of the object 2D location (u_i, v_i) in the image plane at each frame i for $i = 1, \cdots, N$; (2) Estimation of the corresponding 3D locations $\ell_i = (x_i, y_i, z_i)$ constituting the trajectory $\mathcal{Y}$.

7.3 Proposed approach

In order to estimate the 3D trajectory $\mathcal{Y}$, we cast the problem as a 2D trajectory estimation followed by lifting to 3D space [162, 163], where the hypothesis is that temporal information may compensate the lack of the third dimension. This is verified in other applications, e.g., 3D human pose estimation, where the low-dimensional 2D location over time is shown to be discriminative enough to estimate the 3D location with high accuracy [163].
In this section, we describe the main components of the proposed two-step space object 3D trajectory estimation.

7.3.1 2D Location estimation

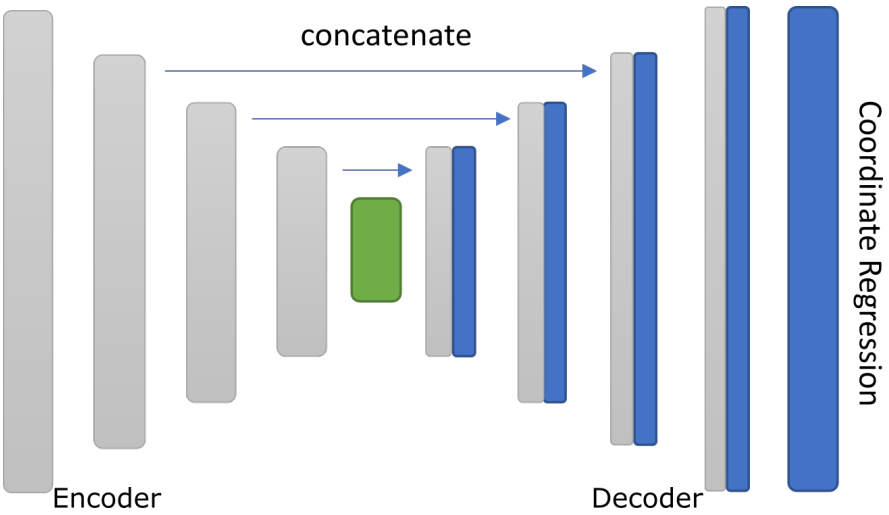


Figure 7.3: **Proposed architecture for 2D point regression**: U-Net [164] is used with ResNet18 [86] as encoder, gray blocks represent the encoder output, green block represent the bottleneck and the blue blocks represent the scaled up output of the decoder. Finally DSNT [165] is used for coordinate regression.

In order to estimate an object 2D location (u, v) from an RGB image I, we represent our object of interest as a single point which is a simpler and a more efficient representation. Indeed, while a common approach is to use a regular bounding box, we choose to track a selected 2D point, i.e., the origin, as it is geometrically related to the desired 3D location

(x, y, z) through Equation (7.1).

Inspired by CenterNet [166], we use an encoder-decoder architecture based on U-Net [164] with ResNet18 [86] as the encoder for feature extraction and the Differentiable Spatial to Numerical Transform (DSNT) [165] to regress the 2D location (u, v), as shown in Figure (7.3). We choose the encoder part of our architecture to be ResNet18 in order to better preserve finer details of the input image especially in the cases where the object is small or far away from the camera. In addition, skip connections are used from the encoder to the corresponding up-convolution in the decoder, and features are concatenated in each corresponding stage between the encoder and the decoder. The final convolution layers of the decoder perceive the spatial resolution of the input image, and output features are passed to a softmax function which produces a single-channel normalized heatmap where all elements are non-negative and sum to one.

This output is passed to the DSNT layer, which is fully differentiable, and exhibits good spatial generalization unlike heatmap matching, and also outputs direct numerical coordinates (u, v).

Then, for a given video sequence $\mathbf{V}_I$, this 2D localization approach:

$$f : I \in \mathbb{R}^{M \times N} \mapsto (u, v) \in \mathbb{R}^2 \tag{7.2}$$

is applied frame by frame on V_I resulting in a sequence of estimated 2D locations $\hat{\mathcal{X}} = \{f(\mathbf{I}_1), \cdots, f(\mathbf{I}_N)\}$. In Equation (7.2), M is the image dimension, and N is the number of frames.

7.3.2 3D Trajectory estimation

Given a sequence $\mathcal{X} \subset \mathbb{R}^2$, the goal is to lift this sequence of 2D locations into the 3D space. To that end, we need to estimate a function $g(\cdot)$, which maps a sequence of 2D points sequence to its corresponding 3D sequence, such that:

$$g : \mathcal{X} \subset \mathbb{R}^2 \mapsto \mathcal{Y} \subset \mathbb{R}^3. \tag{7.3}$$

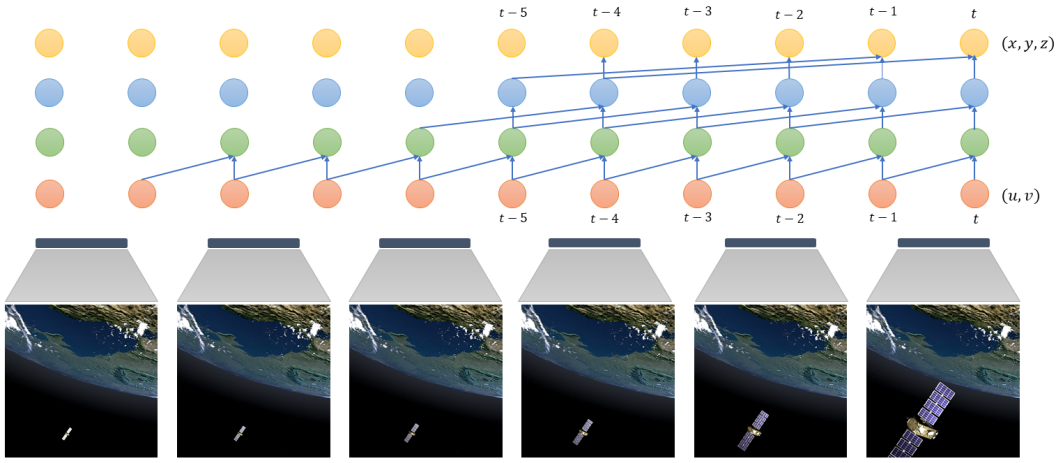


Figure 7.4: 3D Trajectory estimation model for each frame in time t, we forward the historical 2D coordinate (u, v) from the previous frames and estimated its 3D coordinate (x, y, z) using temporal convolution network (TCN) leading to stable and accurate trajectory estimation.

Estimating the 3D location from individual frames leads to a temporally incoherent result, where the independent error from each frame leads to unstable 3D position estimation over the video sequence. Thus, in our work, we follow the same approach proposed in [163, 167] for human pose estimation where a fully convolutional architecture is used to perform temporal convolution over 2D skeleton joint positions in order to estimate the 3D skeleton in a video. Therefore, the function $g(\cdot)$ is approximated by a Sequence to Sequence (Seq2Seq) Temporal Convolutional Network (TCN) model as can be seen in Figure (7.4) using 1D temporal convolution. Consequently, the sequence of 3D locations can be obtained using the combination of the functions $f(\cdot)$ and $g(\cdot)$, such that $\hat{\mathcal{Y}} = g \circ f(V_I)$, where $\circ$ denotes function composition.

We note that TCN is a variation of convolutional neural network for sequence modelling tasks. Compared to traditional Recurrent Neural Networks (RNNs), TCN offers 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 [168]. 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 dilated

convolution [169] layer, 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. The dilated convolution $*_d$ with a dilation factor d of a 1D signal s with a kernel of size k is defined as:

$$(k *_d s)_t = \sum_{\tau=-\infty}^{\infty} k_\tau \cdot s_{t-d\tau}. \tag{7.4}$$

Convolutional models enable parallelization over both the batch and the time dimension while RNNs cannot be parallelized over time [170]. 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 [170]. Architectures with dilated convolutions have been successfully used for audio generation in Wavnet [171], semantic segmentation [172], machine translation [173], and 3D pose estimation [163]. As stated in [170], TCNs generally outperform most of the commonly used networks such as Long Short-Term Memory (LSTM) [174] or Gated Recurrent Unit (GRU) [175] for different tasks.

7.4 Data generation

In the space domain, given the difficulty of obtaining large real datasets, synthetic datasets are currently the default approach for developing DL methods for Space Sitiuational Awareness (SSA) and ADR tasks. To the best of our knowledge, existing datasets [62, 63], and more recently [83], do not provide temporal data as they were designed specifically for single image spacecraft pose estimation [176].

To study spacecraft trajectory estimation, we utilized our realistic space simulation environment, providing a large range of diversity in sensing conditions and trajectories.

Figure 7.5: **Samples from our generated SPARK-T dataset**. Top row – Jason satellite, middle row – heat shield tile, down row – CubeSat.

We used 3D models of three target spacecrafts: (1) a 3D model of *'Jason'* satellite with dimensions $3.8m \times 10m \times 2m$ with the solar panels deployed; (2) 1RU generic *'CubeSat'* with dimensions $10cm \times 11cm \times 11cm$; and (3) for debris we used a heat shield tile model with dimensions $15cm \times 10cm \times 3cm$. The 3D models were obtained from NASA 3D resources [76]. *SPARK-T* dataset was generated by placing the target spacecraft in different trajectories within the field of view of a camera mounted on a chaser. Furthermore, the Sun and Earth were rotated around their respective axes. This has ensured a diversity in the generated dataset with high-resolution photorealistic RGB images for different orbital scenarios.

For this work, 50 sequences were generated for each of the three spacecrafts, with 50 frames each, and including their 3D trajectories as ground truth and the corresponding R, t of the spacecraft with respect to the camera reference frame.

Finally, all images were resized to 512×512 and processed with a zero-mean Gaussian blurring with variance $\sigma^2 = 1$ and an additive Gaussian white noise with variance $\sigma^2 = 0.001$.

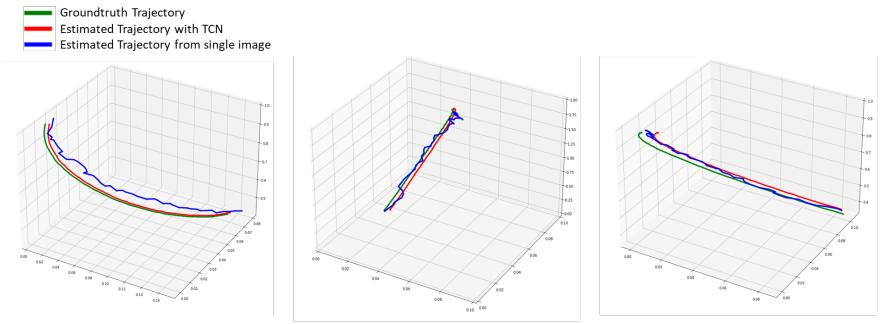


Figure 7.6: **Three examples of groundtruth trajectories:** (in green) and the estimated 3D trajectories using TCN (in red) and the estimated 3D positions using direct regression (in blue).

7.5 Experiments

In this section, we present the experimental setup along with the obtained results. To evaluate the proposed approach, experiments were conducted on our generated spacecraft trajectories dataset presented in Section (7.4).

7.5.1 Data preparation

The data were split into 80% (i.e., 120 sequences) for training , and 20% (i.e., 30 sequences) for testing.

For training the 2D location estimation model $f(.)$ presented in Section (7.3.1), the training data were shuffled in order to eliminate the temporal dependency in the dataset. During training, the input 2D coordinates $p = (u, v)$ were normalized to be in the range [-1, 1], as in [165].

For training the 3D trajectory estimation model $g(.)$, presented in Section (7.3.2), the model was trained with the sequence of 2D location of the ground truth as an input and 3D trajectories ground truth as an output. The 2D / 3D point sequences were normalized in order to have values in the range [0,1] for training the TCN model.

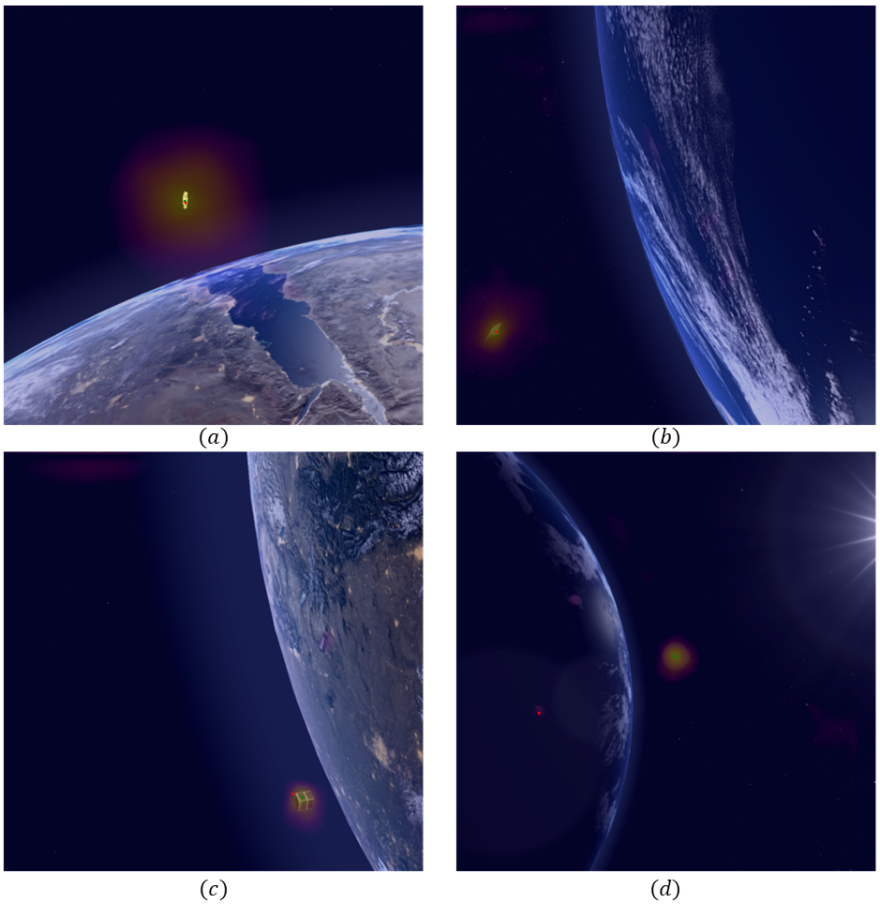


Figure 7.7: **Visualization of the predicted spacecraft 2D location with the heat map overlaid on the input image**. The red point • is the ground truth 2D location, the green point • is the predicted 2D location, in (a) Jason satellite successfully detected, (b) detected debris, (c) detected CubeSat, in (d) wrongly detected CubeSat due to optical sensor sun flare (zooming in might be necessary).

7.5.2 Implementation details

In order to detect the 2D coordinates of the space object present in the image, we train our 2D regression model presented in Figure (7.3) by passing the output of a single-channel normalized heatmap from U-Net to the DSNT layer [165] that outputs numerical coordinates, then we calculate the Euclidean distance[1] between the prediction μ and the ground truth p as

$$\mathcal{L}_{euc}(\mu, p) = \|p - \mu\|_2. \tag{7.5}$$

The estimated 2D coordinate sequences $\hat{\mathcal{X}}$ are passed through a TCN network in order to obtain the corresponding 3D coordinate sequences. By using a TCN network we preserve the temporal coherence present in the 2D sequences which leads, in turn, to improving the quality of the estimated 3D coordinates. The following parameters were used: kernel size $k = 6$; dilation rate $d \in \{1, 2, 4, 8\}$; Adaptive Moment Estimation (ADAM) optimizer with learning rate of 0.001; and 100 epochs.

To highlight the difference between using TCN for temporal consistency and direct 3D location regression, we trained another model similar to the one presented in Figure (7.3) (2D points regression) with adding an auxiliary brunch from the bottleneck of the ResNet encoder to directly regress the 3D location $\ell = (x, y, z)$ and jointly train the model to predict p and ℓ.

7.5.3 Results

We evaluate the obtained results qualitatively and quantitatively at the two levels, namely, (1) 2D location estimation, and (2) 3D trajectory with respect to the camera.
With regards to the 2D location, we have used our proposed model presented in Figure (7.3). As a result we obtained an error $\mathcal{L}_{euc}$ of 0.48 for training and 0.62 for testing. Overall and in most of the cases, the obtained 2D coordinate detection from a single RGB image has a small error. Investigating the cases with high error, we found that those correspond to images generated under direct sun illumination and subject to lens flare. These challenging

[1] No unit as 2D locations are normalized between [-1, 1].

conditions contributed the most to wrongly detected 2D coordinates in these images as can be seen in Figure (7.7) (d). We note, nonetheless, that these frames do not appear continuously in a video. Using multiple frames for estimating the position is therefore a suitable strategy to mitigate errors coming from isolated frames.

In order to estimate the 3D trajectories of the spacecraft, we have lifted the 2D coordinates to 3D space using the proposed TCN based model presented in Section (7.3.2) and illustrated in Figure (7.4). Figure (7.6) shows a visual comparison between the 3D trajectory estimated with the proposed model (red) and the one estimated using direct 3D position regression from single images (blue). We note that our approach provides a smoother and a more temporally coherent trajectory. The overall quantitative result confirms the qualitative observation, with a mean squared error (MSE)[2] of 0.009 for training and 0.012 for testing as compared to 0.084 and 0.174 for training and testing, respectively, in the case of direct 3D position regression. The obtained results confirm a significant improvement as compared to directly estimating the 3D locations from the corresponding RGB images.

7.6 Conclusion

In this Chapter, we investigated the problem of spaceobject 3D trajectory estimation using only RGB information. We proposed a two-step approach decomposing the problem into: (1) a per-image 2D spacecraft detection; followed by (2) a per-sequence 3D trajectory estimation. Our experimental results showed that by properly leveraging temporal information, it is possible to simplify the problem and further increase accuracy as compared to a direct 3D position regression. Furthermore, we proposed a large realistic synthetic dataset that provides ground truth trajectories for three spacecrafts, under various sensing conditions. This dataset will be publicly shared with the research community in order to further the research on spacecraft trajectory estimation in the context of ADR.

[2]No unit as coordinates are normalized.

Chapter 8

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

This chapter 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.

8.1 Introduction

With the increase in the number of space missions and debris [52, 8, 177], 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 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 [178]. 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 [84]. 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) [179] – 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 [179]. As a consequence, existing spaceborne images captured from previous missions are sometimes combined with synthetic data. In particular, the Cygnus dataset [184] contains 540 pictures of the Cygnus spacecraft in orbit in conjunction with 20k synthetic images generated with Blender [187]. However, the main limitation of spaceborne

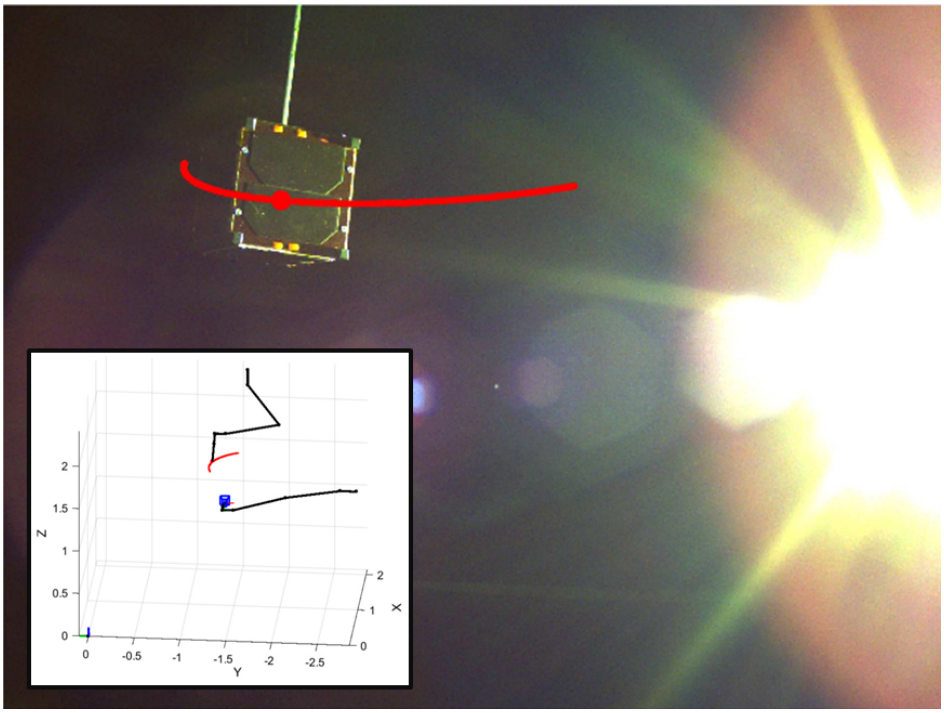


Figure 8.1: **Illustration of the SnT Zero-G Laboratory data**. Top: chaser camera field of view featuring a CubeSat target with its projected trajectory (in red). Bottom left: instantaneous 3D positions of the robotic arms simulating the chaser (camera in blue) and target (trajectory in red) spacecrafts.

images is 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 [181]. Moreover, laboratories offer monitoring mockup poses and environmental conditions to ensure a higher quality of the data [188, 75]. 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 [181] 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 [91],

	SPARK [180]	SPEED [179]	SPEED+ [181]	URSO [182]	SwissCube [183]	Cygnus [184]	Prisma12K [185]	Prisma25 [186]	**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 8.1: Overview of existing SSA datasets. In the table, *sim* refers to simulated images and *lab* to laboratory data.

189], 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 work, 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 [180, 176, 179, 181, 182, 183, 184, 185, 186]. Indeed, estimating the pose – especially orientation – of a symmetric object is a difficult task that has been receiving interest from the research community [190, 191, 192]. 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, referred to as *CubeSat-CDT*, where "CDT" stands for Cross-Domain Trajectories.

- A novel algorithm for spacecraft trajectory estimation, built upon previous work [193], 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 work is organised as follows. Section (8.2) provides a comparative review of existing SSA datasets. Section (8.3) presents the proposed method to leverage the temporal information for trajectory estimation and cross domain data validation. Section (8.4) presents an analysis of the performance of our proposed approach on the CubeSat-CDT dataset.

8.2 Related datasets

Multiple datasets are provided to address the SSA challenge. Some are freely available for research [180, 179, 181, 182, 183], others are proprietary from some space companies and are not publicly available [184, 185, 186]. These datasets are discussed below, highlighting their contributions, limitations and difference with CubeSat-CDT. Table (8.1) summarizes this study.

URSO [182] provides 15k images of two different targets, the 'Dragon' spacecraft and the 'Soyuz' one, with different operating ranges and at a resolution 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 [183] 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 [184] 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 [185] 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 [186] 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 Chapter (3) Section (3.6), contains multiple trajectories in three different domains with real spacecraft - see Table (8.1) - . We believe such a dataset opens new possibilities for studying trajectory estimation of spacecrafts.

8.2.1 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 Figure (3.6), 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. Figure (8.2) presents the position and orientation of the trajectories performed in the three datasets. As summarized in Table (3.1), 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.

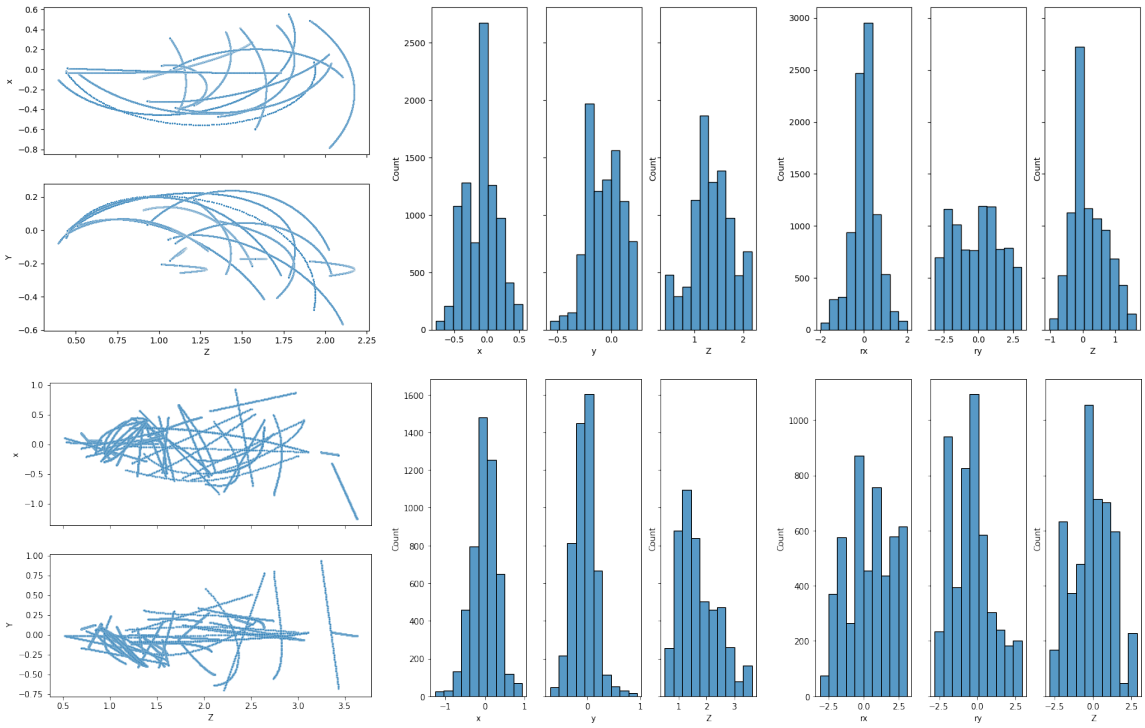


Figure 8.2: **Trajectory analysis for the Blender & Zero-G Lab (First row) and SPARK (Second row) subsets**. From left to right: trajectory analysis in x and y vs z axis, range distribution for all the axes, angle distribution for all the axes.

8.3 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.

8.3.1 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 [193], let $V_I = \{I_1, \cdots, I_N\}$ be a sequence of RGB images featuring the observed spacecraft, N being the total 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.

8.3.2 Proposed Approach

The proposed model is composed of an EfficientNet B2 [194] 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 [194] feature extractor,

$$f : I \in \mathbb{R}^{M\times N} \mapsto Z \in \mathbb{R}^{\Psi}, \tag{8.1}$$

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 (8.1), $\Psi = 128$ is the dimension of the extracted CNN features, M is the image dimension, and N is the number of frames.

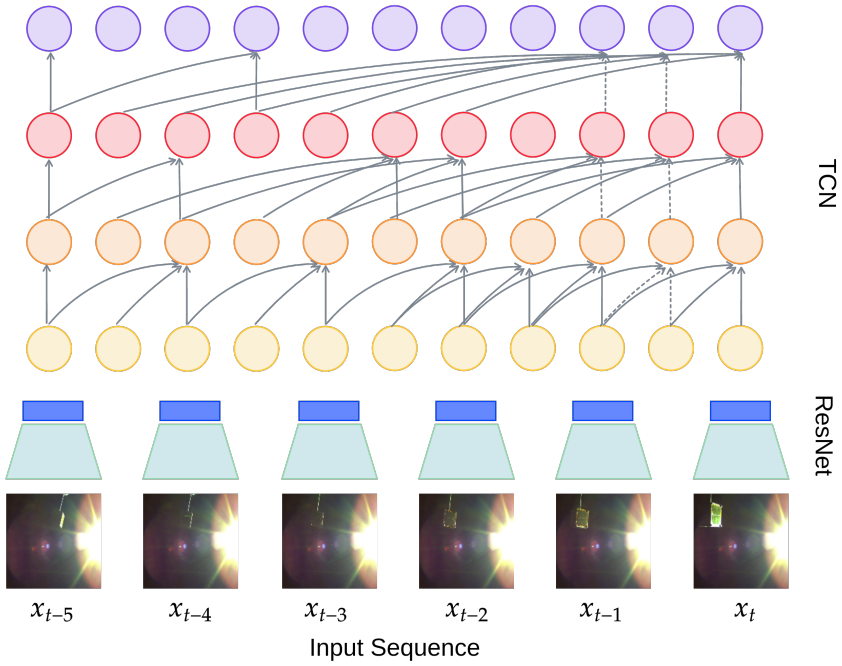


Figure 8.3: **Our TCN-based trajectory estimation model**. At time t, we leverage the spatial features of each frame to estimate the target poses using a TCN, leading to smooth and accurate trajectory estimation.

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:

$$g : \mathcal{X} \in \mathbb{R}^{\Psi \times N} \mapsto \mathcal{Y} \in \left(\mathbb{R}^3 \times \mathrm{SO}_3(\mathbb{R})\right)^N. \tag{8.2}$$

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 Figure (8.3).

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

$$\hat{\mathcal{Y}} = g \circ f(V_I). \tag{8.3}$$

8.3.3 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 [168]. 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 dilated convolution layer [169], 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 [170]. 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 [170].

8.4 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 (8.3), 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 Figures. (8.4) and (8.5).

We use a PoseNet model [195] with an EfficientNet [194] backbone for feature extraction,

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 8.2: 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 8.3: Pose MSE when regressed by the Temporal Convolutional Network for the three data domains.

followed by fully connected layers for pose regression.

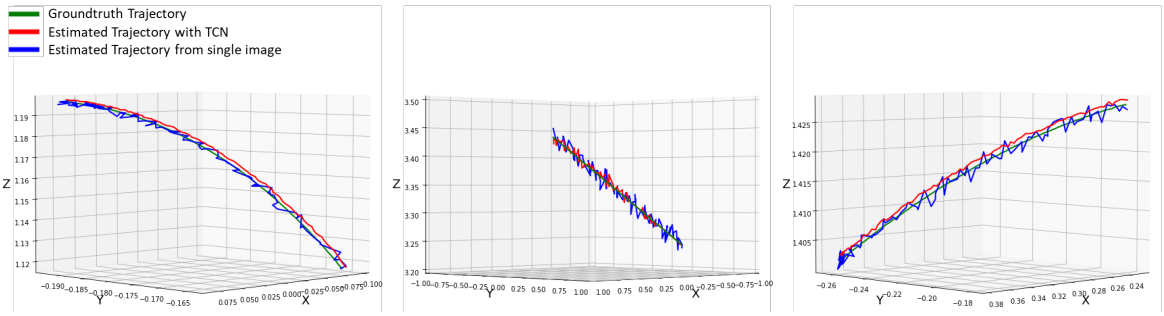


Figure 8.4: Groundtruth trajectories of the CubeSat center (in green), estimated trajectories using TCN (in red) and estimated positions using single-frame regressions (in blue). From left to right: example sequences from Zero-G Lab, SPARK and Blender.

8.4.1 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 information is not used to evaluate the

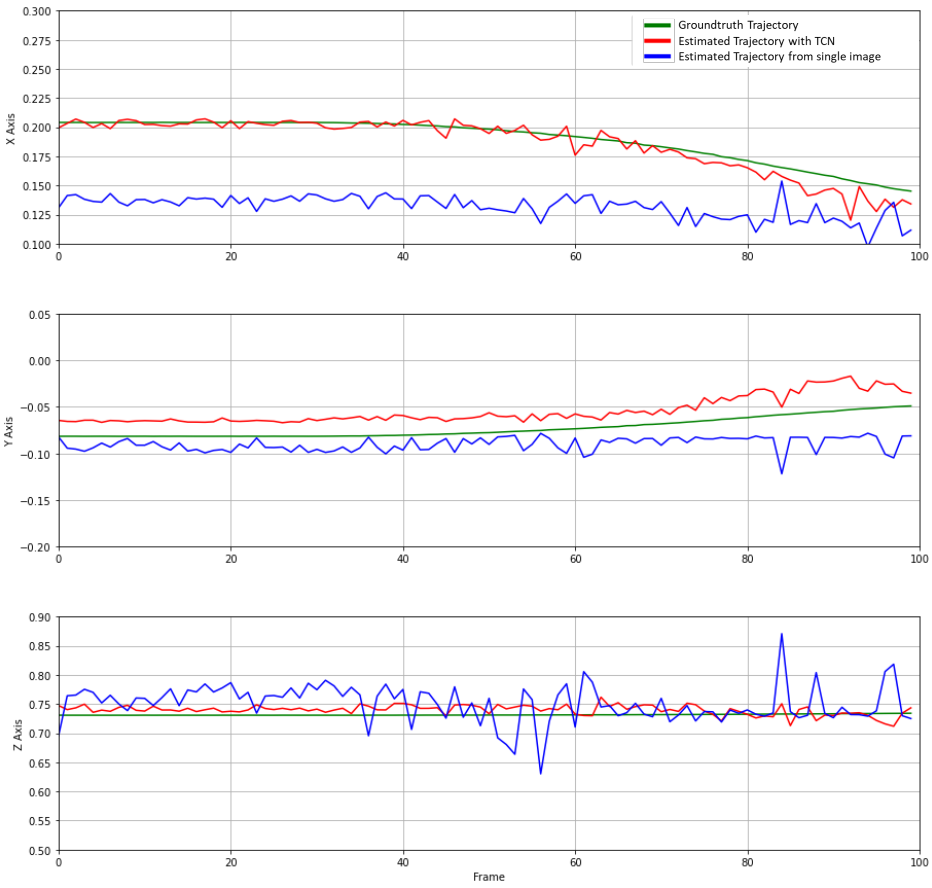


Figure 8.5: Per Axis position estimations for *Sequence*7 from Zero-G Laboratory data.

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 (8.2), 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.

8.4.2 Impact of Temporal Information

The second experiment was designed to assess the impact of temporal information on pose estimations. The model proposed in Section (8.3) 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.

8.5 Conclusions

In this work, 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.

Chapter 9

Self-Supervised Learning for Place Representation Generalization across Appearance Changes

This chapter addresses the challenge of visual place recognition, focusing on the application of self-supervised learning to acquire image features that are robust to appearance changes and sensitive to geometric nuances. This methodology demonstrates strong visual place recognition capabilities under varying seasonal and illumination conditions, all achieved without the need for human-annotated labels.

The development of these appearance-robust and geometry-sensitive features is particularly relevant for OOS systems. In the unique and often unpredictable space environment, these systems require advanced visual recognition capabilities to navigate and operate effectively. The self-supervised learning approach, therefore, holds significant promise for enhancing the autonomy and reliability of OOS systems in space.

Visual place recognition is essential for spatial navigation across various entities, including animals, humans, and robots. Traditional state-of-the-art methods, typically reliant on supervised learning, often struggle to generalize to atypical conditions due to their training limitations. However, self-supervised learning offers a promising alternative, potentially

enabling the abstraction of place representations to be more adaptable across diverse conditions. In this context, and linking back to the previous discussion on visual place recognition for OOS systems, this chapter investigates the development of image features that are resilient to changes in appearance while remaining sensitive to geometric alterations, using self-supervised learning techniques.

This approach combines the core principles of self-supervised learning, namely contrastive and predictive learning, to achieve this dual objective. The efficacy of this method is demonstrated through our results on standard benchmarks, which show that such a combined learning approach leads to competitive outcomes in visual place recognition. These results are particularly significant under challenging seasonal and illumination conditions and are achieved without the dependency on human-annotated labels, underscoring the potential of self-supervised learning in enhancing the capabilities of OOS systems in the dynamic space environment.

9.1 Introduction

Visual Place Recognition (VPR) is central for localizing - *i.e.* determining a camera's position in a scene [196, 197], and has applications from autonomous driving to augmented reality. Typically viewed as an image retrieval task, VPR aims to match a *query* image to images in a *reference* database that depict the same location, even when conditions like viewpoint, obstructions, or weather vary [198]. This makes VPR challenging but vital for dependable real-world vision-based systems.

For this goal, neuroscience research indicates that biological intelligence relies on creating abstract representations of places, known as *cognitive maps* [198], to recognize them under varying conditions [199].

These maps are essential for generalizing limited knowledge, such as recognizing a place seen only in daylight during nighttime. The aim is to build rich representations reflecting the intrinsic structures that are not required to be re-learned from scratch when non-critical visual information changes [199].

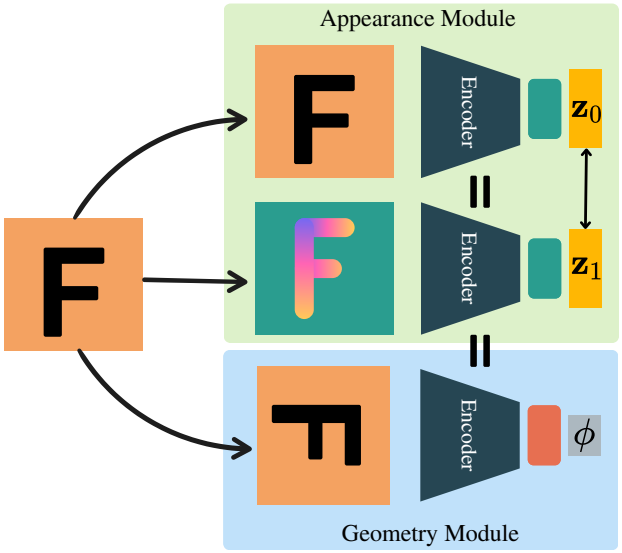


Figure 9.1: **CLASP-Net Training Strategy:** Three views are generated from an input image. The Appearance Module in green (top) maps the original and appearance-augmented views into close representation vectors $\{\mathbf{z}_0, \mathbf{z}_1\}$. The Geometry Module in blue (bottom) predicts the transformation ϕ applied between the original and third views.

In the context of technological solutions, state-of-the-art VPR methods have focused on achieving invariance to both environmental conditions and viewpoint changes in image representations. The latter are for recognizing places observed under unprecedented angles [200, 201]. However, we argue that such viewpoint invariance may be detrimental in the process of distinguishing between different places. Moreover, recent works have shown that favouring a more general equivariance in image representations may be more beneficial than seeking *only* invariance [202, 203, 204].

Motivated by this, we introduce **CLASP-Net** :*Contrastive Learning with Appearance Augmentations and Spatial Predictions for Place Recognition*[1], designed to learn discriminative place representations that can generalize to new conditions. We use Self-Supervised Learning (SSL) to address training limitations due to low appearance variability in reference images. Contrastive Learning (CL) is employed to unify representations of the same place

[1]Clasp: [noun] a device, usually of metal, for fastening together two or more things or parts of the same thing.

by using appearance augmentations. To further exploit the scene's spatial layout and regularize the model, we apply geometric transformations and utilize a Predictive Learning (PL) framework for classification based on these transformations.

Contrary to supervised learning, which tends to learn shortcuts and struggles to generalize from limited labelled data [205], SSL seems closer to human-like learning and does not require manual annotation [206]. Few studies have explored SSL for VPR [207, 208]. We propose to merge the key SSL approaches, CL [209] and PL [210], aiming for image representations that are resilient to appearance shifts while being sensitive to geometric cues. By doing that, we aim to learn features suitable for visual place recognition under appearance changes.

Contributions. Our contributions are two-fold:

(1) A novel approach for Visual Place Recognition under extreme condition changes, CLASP-Net, that leverages both contrastive and predictive self-supervised learning approaches.

(2) An experimental evaluation confirming the competitiveness of CLASP-Net compared to state-of-the-art approaches on standard benchmarks featuring different conditions (day/night, weather, seasons), among which the very challenging Alderley Dataset [211].

Chapter organization. The rest of the chapter is organized as follows. Relevant work on SSL and VPR is reviewed in Section (9.2). CLASP-Net is presented in Section (9.3), while experimental evaluation demonstrating the validity of our approach is reported in Section (9.4). Section (9.5) concludes the chapter and presents future works.

9.2 Related Work

9.2.1 CNN-based Descriptors for Visual Place Recognition

The rapid evolution of deep learning has opened new avenues for overcoming the limitations of traditional, handcrafted descriptors. Following the groundbreaking work by Chen *et al.*[212], there has been a growing emphasis on learning-based descriptors primarily built from Convolutional Neural Networks (CNNs). For instance, Sunderhauf *et al.* [213] and Hou *et al.* [214] found that mid-level features from trained CNN model are more resilient to variations in appearance.

Moreover, a concerted effort has been made to design specialized neural networks for VPR tasks. This has led to the invention of techniques like CALC [215], NetVLAD [216], NetBoW [217] and NetFV [218] that meld the best aspects of both traditional and learning-based descriptors, achieving unprecedented results.

In terms of performance, CNN-based descriptors, particularly those relying on supervised learning, are highly dependent on extensive, high-quality training datasets.

However, it's crucial to acknowledge that supervised learning methods often require laborious data annotation, which can be both time-consuming and costly. Therefore, self-supervised learning presents a compelling alternative to VPR tasks.

9.2.2 Self-Supervised Learning

Self-supervised methods focus on learning visual features from large sets of unlabeled images, making them valuable for diverse real-world applications such as autonomous driving. These methods usually employ a pretext task with a related objective function for training [210]. The objective function can target either network predictions (predictive learning) or the feature representation space (contrastive learning). This enables SSL to yield image representations that are both sensitive and robust to specific transformations.

Predictive Learning. PL uses pretext tasks to indirectly infuse image representations with inductive biases via network outputs [210]. Tasks range from image colorization [219] and jigsaw puzzles [220] to rotation prediction [221]. These tasks encourage the network to learn rich object representations and their spatial arrangements. For instance, predicting an outdoor scene often involves recognizing sky and trees at the top and roads at the bottom, requiring an understanding of the scene's structure.

Contrastive Learning. CL directly refines image representations using a contrastive loss that considers batch elements' relationships. SimCLR [209], a framework for visual representation through CL, stands out for its simplicity, not needing specialized structures [222] or memory banks [223, 224]. It works by sampling two distinct augmentations, applying each

to an image, and then training encoders on a contrastive loss to maximize similarity between the two views and minimize similarity with different images. To address potential training convergence challenges in CL, ScatSimCLR [225] also estimates each view's augmentation parameters.

Combining Predictive and Contrastive Learning. CL aims at inducing invariance to some content-preserving transformations while being distinctive to such content changes. On the other side, PL is mostly used to incorporate sensitivity, and ideally equivariance, to given transformations into representations [202]. Some studies have demonstrated the advantage of balancing invariance and equivariance [226, 227, 228]. For example, Winter et al.[229] suggested an AutoEncoder-centric framework to cultivate representations that exhibit both robustness and sensitivity to rotations. Explicitly, an encoder translates a rotated image into a more invariant latent representation, from which a decoder predicts the unrotated original image. Simultaneously, an auxiliary branch pursues equivariance by determining the rotation angle. In a similar vein, Feng *et al.* [230] endeavour to learn features impervious to the rotation of input images by bifurcating the features: one segment is dedicated to rotation prediction (dubbed equivariant features), while another segment, subjected to a contrastive loss, penalizes disparities emerging from various rotations (termed invariant features).

In recent work, Dangovski *et al.* [203] introduced Equivariant Self-Supervised Learning (E-SSL), a more nuanced SSL approach that goes beyond simply seeking invariant representations. E-SSL framework enriches traditional SSL methods by integrating both equivariance and invariance objectives in the pre-training process. The key insight is that some transformations are better captured as equivariant, meaning that the learned features should change predictably based on how the input is transformed. At the same time, other transformations are better captured as invariant, where the feature representation should remain constant despite changes of the input.

Drawing on these insights, our proposed CLASP-Net focuses on achieving appearance invariance through CL while capturing detailed representations of scene components and their spatial layouts through PL. With the latter, the network gains sensitivity to geometric

transformations, enhancing its suitability for VPR tasks.

9.2.3 Self-Supervised Learning for Visual Place Recognition

As highlighted in Section (9.2.2), SSL is well-suited for VPR because it addresses the issue of unrepresentative training data due to varying test conditions. Despite its promise, few methods exist. For instance, Tang *et al.* [208] have proposed to disentangle appearance-related and place-related features using a generative adversarial network with two discriminators. However, this type of method may suffer from unstable training. SeqMatchNet [207] is a CL-based method that leverages sequences of video frames in the contrastive loss to robustify image representations for VPR.

From a larger perspective, Mithun et al. [231] use sets of related images (i.e., showing the same place under different conditions) to enhance VPR image representations. Thoma et al. [232] suggest loosening geo-tag constraints for weakly-supervised training. Unlike these works, we generate pairs of corresponding images in a self-supervised manner, without labels. Venator et al. [47] employ SSL to create appearance-invariant descriptors for image matching, which could serve as a refinement step in our approach.

9.3 Proposed CLASP-Net

Our primary objective is to enable the model to learn features that can withstand drastic changes in appearance while remaining effective for VPR. Specifically, we aim to create image representations that capture essential geometric details of the scene's spatial arrangement yet remain unaffected by varying environmental conditions. To accomplish this, we integrate both sensitivity to geometric information and robustness to appearance changes into the image representations using self-supervised learning techniques.

9.3.1 Problem Formalization

Following the traditional approach [198], we frame the VPR problem as an image retrieval task, where, given a query image $\mathbf{q}$ depicting a place $\mathscr{P}_{\mathbf{q}}$, a representation *a.k.a.* descriptor

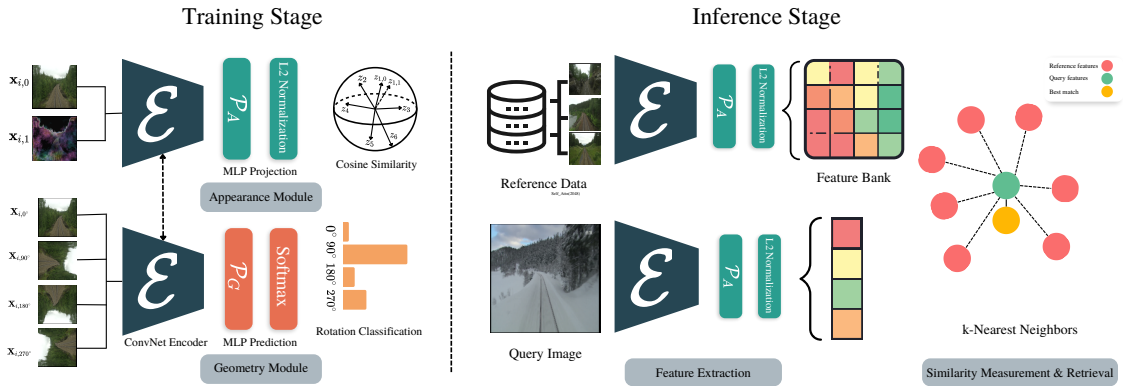


Figure 9.2: Overview of CLASP-Net. **Training Stage:** from an original image $\mathbf{x}_{i,0}$, augmented versions with a modified appearance $\mathbf{x}_{i,1}$ and different orientations ($\mathbf{x}_{i,0°}, \mathbf{x}_{i,90°}$, $\mathbf{x}_{i,180°}$, $\mathbf{x}_{i,270°}$) are generated. Representations of the first two images are brought closer thanks to a contrastive learning framework to achieve appearance robustness. In parallel, original and rotated images are passed through a classification network sharing the same encoder to predict the applied transformation and achieve geometric sensitivity. Note that our method does not rely on any manual annotation. **Inference Stage:** The representations from query and reference images are compared based on similarity measure then the closest k reference images constitute the image retrieval output.

$\mathbf{z_q}$ of that image is computed. It is then compared to the descriptors $\{\mathbf{z}_i\}_{i=1..N_R}$ of reference images $\{\mathbf{x}_i\}_{i=1..N_R}$, where N_R is the size of the reference database. The comparison is done using a given similarity metric (*e.g.*, cosine similarity). This inference stage is illustrated in Figure (9.2).

During the training, the model only has access to reference images that we assume unlabelled. Moreover, the environmental conditions under which the query image is acquired are not necessarily similar to the ones featured in the reference database, making the problem very challenging, even sometimes for human eyes.

9.3.2 Preliminaries: Robustness & Sensitivity

Our approach focuses on extracting image features that are both robust to appearance changes and sensitive to geometric aspects. Mathematically, these properties correspond to the concepts of invariance and equivariance. Formally, let $\mathfrak{G}$ be a generic group of transformations and $\mathfrak{g}$ an element of $\mathfrak{G}$. The actions of $\mathfrak{g}$ on the input and output spaces of a function

$\mathcal{F} : \mathbb{I} \rightarrow \mathbb{O}$ are denoted by $\phi_{\mathfrak{g}}^{(\mathbb{I})}$ and $\phi_{\mathfrak{g}}^{(\mathbb{O})}$, respectively.

In practice, considering an encoder model $\mathcal{E}$ for extracting features from an image $\mathbf{x}$, we seek robustness to any appearance transformation $\mathcal{T}_A$:

$$\forall \mathcal{T}_A, \forall i \in [1; N_R], \quad \mathcal{E}(\mathcal{T}_A \mathbf{x}_i) \approx \mathcal{E}(\mathbf{x}_i), \tag{9.1}$$

while, at the same time, sensitivity to a certain group of geometric transformations $\mathfrak{G}_G$:

$$\forall \mathcal{T}_G \in \mathfrak{G}_G, \forall i \in [1; N_R], \quad \mathcal{E}(\mathcal{T}_G \mathbf{x}_i) \approx \mathcal{T}'_G \mathcal{E}(\mathbf{x}_i), \tag{9.2}$$

where $\mathcal{T}'_G \approx \mathcal{T}_G$. The different possible groups of transformations are investigated in Section (9.4).

9.3.3 Model Architecture

Our pipeline exploits both CL for encouraging invariance to appearance changes and PL for encouraging sensitivity to geometric image augmentations. This hybrid approach is consistent with the *E-SSL* framework proposed in [203]. The overall architecture of the proposed CLASP-Net is presented in Figure (9.2).

At training time, CLASP-Net is composed of two branches sharing the weights of an encoder model $\mathcal{E}$. The first branch, denoted *Appearance Module*, takes as inputs the original image $\mathbf{x}_i$ and an augmented version with modified appearance $\mathcal{T}_A \mathbf{x}_i$, then applies a contrastive learning loss in the representation space to bring the two descriptors closer. The second branch, denoted *Geometry Module*, uses rotated versions of the original image, $\mathcal{T}'_G \mathbf{x}_i = \mathrm{R}(n^\circ)\mathbf{x}_i$, and predicts the angle of the rotation n.

Appearance Module. The first branch, divided into two sub-branches (see Figure (9.1)), This setup is inspired by SimCLR [209] and employs a shared encoder $\mathcal{E}$ and MultiLayer Perceptron (MLP) $\mathcal{P}_A$ mapping between the image domain and the latent representation space where the contrastive loss is applied. Given original images $\mathbf{x}_i$ along with their augmented versions $\mathcal{T}_A \mathbf{x}_i$, the weights of the two networks are learned using a contrastive loss. This loss, formalized in Section (9.3.4), ensures that the descriptor of each version, *e.g.*, $\mathcal{E}(\mathcal{P}_A(\mathbf{x}_i))$,

is similar to the descriptor of its corresponding view, $\mathcal{E}(\mathcal{P}_A(\mathcal{T}_A\mathbf{x}_i))$, while distant from the other descriptors. The intuition behind this module is to force the encoder model $\mathcal{E}$ to learn features agnostic on the conditions (e.g. illumination, weather, season) under which the place was initially observed.

Geometry Module. The second branch incorporates the same shared encoder $\mathcal{E}$ along with a prediction-focused $\mathcal{P}_G$. This setup is designed to classify rotated versions of the original image, denoted $\mathrm{R}(n°)\mathbf{x}$, based on their rotation angle n. Utilizing a standard cross-entropy loss, the module aims to train the encoder $\mathcal{E}$ to learn rich representations of scene layout and spatial arrangement and capture geometry-sensitive features vital for accurate place recognition.

Combined, these two modules work together to disentangle appearance and geometric aspects of input images, enabling robust visual place recognition even when appearance conditions vary. During inference, the architecture used to compute image descriptors consists of the encoder $\mathcal{E}$ followed by the projector network $\mathcal{P}_A$, as shown in Figure (9.2) (right part).

9.3.4 Model Loss

Figure 9.3: **Examples of augmentations leveraged by CLASP-Net**. Top row (a): an original input batch from Oxford RobotCar v2 dataset, (b) pixel-level augmentations for appearance changes, (c) random rotations applied on the original image.

Note: For the sake of clarity, we herein introduce more specific notations for denoting images

and their augmented/rotated versions.

We use a combination of contrastive and predictive losses to steer our model toward robustness to appearance changes and sensitivity to geometric variations.

Given a random batch of N reference images $\mathcal{B} = \{\mathbf{x}_{i,0}\}_{i=1..N}$ corresponding to N different places, we apply one random appearance transformation to each image. By so doing, we create N additional images $\{\mathbf{x}_{i,1}\}_{i=1..N}$. These $2N$ images constitute the contrastive batch $\mathcal{B}_C = \{\mathbf{x}_{i,j}\}_{i=1..N,j\in\{0,1\}}$ that is fed into the Appearance Module. Furthermore, we also apply rotations of 0°, 90°, 180° and 270° to each original image. As a result, we create the predictive batch of $4N$ images $\mathcal{B}_P = \{\mathbf{x}_{i,j^\circ}\}_{i=1..N,j\in\Theta_4}$, where $\Theta_4 = \{0°, 90°, 180°, 270°\}$. $\mathcal{B}_P$ is fed into the Geometry Module.

Contrastive loss. The contrastive batch $\mathcal{B}_C$ contains N *positive* pairs of images $(\mathbf{x}_{i,0}, \mathbf{x}_{i,1})$ depicting the same place, the rest being *negative* pairs corresponding to different places. We use NT-Xent loss [209] that leverages positive samples, and is based on the cosine similarities between the obtained image representations $\mathbf{z}_{.,.} = \mathcal{P}_A(\mathcal{E}(\mathbf{x}_{.,.}))$, expressed as

$$\mathrm{s}(\mathbf{z}_{i,j}, \mathbf{z}_{k,l}) = \frac{\mathbf{z}_{i,j} \cdot \mathbf{z}_{k,l}}{\|\mathbf{z}_{i,j}\|\|\mathbf{z}_{k,l}\|}, \tag{9.3}$$

where $\cdot$ is the dot product.
Specifically, the contrastive loss is defined as

$$\mathcal{L}_C = \frac{1}{2N}\sum_{i=1}^{N} \ell_{0\to1}(i) + \ell_{1\to0}(i), \tag{9.4}$$

where

$$\ell_{a\to b}(i) = -\log\frac{\exp(\mathrm{s}(\mathbf{z}_{i,a}, \mathbf{z}_{i,b})/\tau)}{\sum_{k=1}^{N} \mathbb{1}_{k\neq i} \sum_{j=0}^{1} \exp(\mathrm{s}(\mathbf{z}_{i,a}, \mathbf{z}_{k,j})/\tau)}, \tag{9.5}$$

with τ denoting a temperature parameter that controls the strength of penalties on pairs of non-corresponding images [118] and $\mathbb{1}_{k\neq i}$ being equal to 1 if $k \neq i$, and 0 otherwise.

The contrastive loss aims at making representations of the same place under different conditions similar to each other, while forcing representations of different places to be different.

Data Augmentation Type	Probability
Planckian Jitter	0.8
Color Jiggle	0.5
Plasma Brightness	0.5
Plasma Contrast	0.3
Gray scale	0.3
Box Blur	0.5
Channel Shuffle	0.5
Motion Blur	0.3
Solarize	0.5

Table 9.1: List of data augmentations applied to the images on-the-fly during training. We also set a probability for each one of them.

Predictive loss. The predictive batch $\mathcal{B}_\mathcal{P}$ contains four rotated views of each place. The task of this branch is to predict the rotation angle for each of the $4N$ pictures. We frame this as a classification problem with 4 classes corresponding to 0°, 90°, 180° and 270° rotation angles. The predictive loss is therefore the standard cross-entropy loss:

$$\mathcal{L}_P = -\sum_{i=1}^{N} \sum_{j \in \Theta_4} \mathrm{c}(\mathbf{x}_{i,j}) \cdot \log(\widetilde{\mathbf{z}}_{i,j}), \tag{9.6}$$

where $\widetilde{\mathbf{z}}_{i,j} = \mathrm{Softmax}(\mathcal{P}_G(\mathcal{E}(\mathbf{x}_{i,j}))) \in \mathbb{R}^4$ is the prediction, log() the element-wise natural logarithm, $\cdot$ the dot product and $\mathrm{c}(\mathbf{x}_{i,j}) \in \mathbb{R}^4$ the groundtruth with elements equal to 0 except the nth element equal to 1 if the true rotation is $(n-1) \times 90°$.

Overall loss. The final loss is the combination of the contrastive loss for appearance robustness and predictive loss for geometry sensitivity:

$$\mathcal{L} = \mathcal{L}_C + \lambda.\mathcal{L}_P, \tag{9.7}$$

where λ is a weighting factor to balance the two terms.

9.4 Experimental Evaluation

9.4.1 Datasets

The Nordland dataset [233]: records a 728 km long train journey connecting the cities of Trondheim and Bodø in Norway. It contains four long traversals, once per season, with diverse visual conditions. The dataset has 35768 images per season with one-to-one correspondences between them. We follow the dataset partition proposed by Olid *et al.* [234] with test set made of 3450 photos from each season.

The Alderley dataset [211]: records an 8 km travel along the suburb of Alderley in Brisbane, Australia. The dataset contains two sequences: the first one was recorded during a clear morning, while the second one was collected on a stormy night with low visibility, which makes it a very challenging benchmark. The dataset contains 14607 images for each sequence and each place have 2 images. We train our approach on the day sequence and test on the night sequence.

The Oxford RobotCar Seasons v2 dataset [235]: is based on the RobotCar dataset [236], which depicts the city of Oxford, UK. It contains images acquired from three cameras mounted on a car. There are 10 sequences corresponding to 10 different traversals carried out under very different weather and seasonal conditions. The rear camera images of the *overcast-reference* traversal (6954 images) are used as a basis for reference training images, to which we add 1906 rear camera images from other traversals following the *v2* train/test split. These additional images cover different environmental conditions but only a subset of places (not full traversals). The test set contains 1872 images from all traversals except *overcast-reference*, without overlap with training images.

9.4.2 Evaluation

The evaluation on both Nordland and Alderley datasets uses the recall R@N measure, which consists in the proportion of successfully localized query images when considering the first N retrievals. If at least one of the top N reference images is within a tolerance window around the query's ground truth correspondence, the query image is deemed succesfully localized.

Method	Nordland Summer/Winter		
	R@1	R@5	R@10
NetVLAD [216]	7.7	13.7	17.7
SFRS [237]	18.8	32.8	39.8
SuperGlue [238]	29.1	33.5	34.3
DELG [239]	51.3	66.8	69.8
Patch-NetVLAD [240]	46.4	58.0	60.4
TransVPR [241]	**58.8**	**75.0**	*78.7*
CLASP-Net (Ours)	*53.0*	*73.8*	**80.2**

Table 9.2: Quantitative results on Nordland dataset. Best results are in **bold**. Second best results are in *italic*.

Method	Alderley Day/Night
NetVLAD [216]	3.35
CIM [242]	7.82
Patch-NetVLAD [240]	7.99
Seqslam [211]	9.90
Retrained NetVLAD [243]	15.8
AFD [243]	21.0
CLASP-Net (Ours)	**25.2**

Table 9.3: Quantitative results on Alderley dataset. Best result is in **bold**.

The tolerance window is set to two frames distant from the query before and after, so that the window contains 5 pictures. Following the common approach for NordLand [244, 245, 240], images of the winter sequence are used as queries, while the summer sequence is used as reference.

For RobotCar-Seasons v2, we follow the Patch-NetVLAD [240] approach and utilize the 6-DoF pose of the best-matched reference picture as prediction of the query's pose. Since we don't compute any pose, our image retrieval method is not comparable with pose estimation methods such as MegLOC [246].

9.4.3 Implementation details

Encoder model $\mathcal{E}$. We use ResNet50 [125] as the backbone, with pre-training on ImageNet using the Timm library [247]. The last classification layer is discarded so that the model is only used for the feature extraction.

Rotation predictor $\mathcal{P}_G$. We use a simple 1-layer perceptron with layer normalization and ReLU activation.

Projector $\mathcal{P}_A$ We use a simple 1-layer perceptron with batch normalizations and ReLu activation. The dimension of the output (*i.e.*, image descriptor) is 1024.

Appearance Augmentations. Following domain generalization approaches, our model leverages numerous pixel-level data augmentations to trigger appearance invariance bias in the model. The list of pixel-level augmentations for appearance modification is provided in Table (9.1), while examples of such augmentations are provided in Figure (9.3). The chosen set of variations empirically achieved good performance whereas other tested combinations were less favourable. We use the Kornia [248] library for self-supervised data augmentation.

Geometric Augmentations. Our training strategy encourages information about rotations to be retained in the image representation rather than guaranteeing strict equivariance. Moreover, the choice of this particular group of geometric transformations is the outcome of experimentations whose results are presented in Figure (9.4). Empirically, we found that the best performance is achieved with the cyclic group of 90° rotations, compared to the groups of 2D affine transformations, 2D projective transformations, and 2D rotations.

Model training. The model is trained for 1000 epochs using Adam optimizer [249] and a batch size of 64. Although contrastive learning usually requires larger batch size [250], using Adam optimizer allowed us to obtain good results with a smaller batch size. A learning rate of 0.003 had the best performance with this optimizer. The temperature parameter τ is set to 0.01 and the loss factor λ is set to 1 in our experiments.

Inference. Prior to the inference stage, we pass the set of reference images to the Appearance Invariant Module of the trained model: $\mathcal{E} \rightarrow \mathcal{P}_A \rightarrow L2-$normalization and thus build a reference descriptor bank. A k-Nearest Neighbor search based on cosine similarity to find the closest references to the query image.

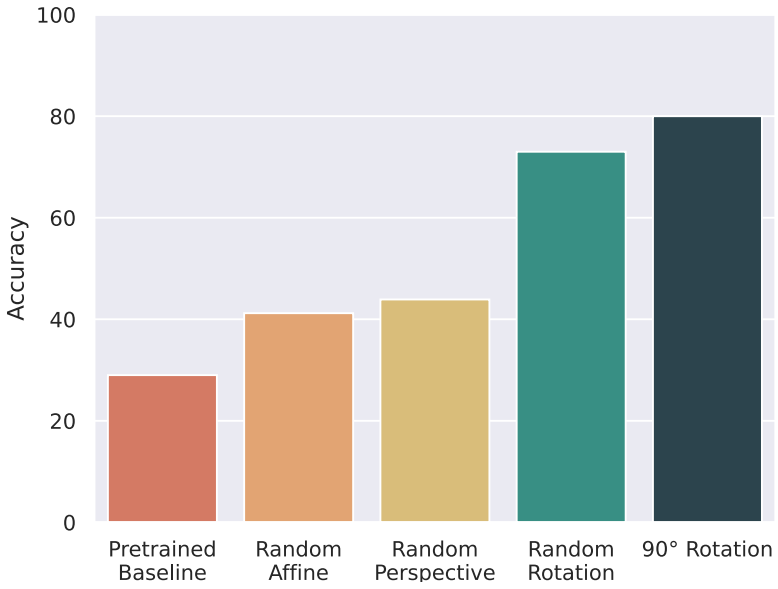


Figure 9.4: R@10 on Nordland Summer/Winter dataset with Geometry Modules relying on different groups of transformations.

9.4.4 Results

	day conditions							night conditions	
	dawn	dusk	OC-summer	OC-winter	rain	snow	sun	night	night-rain
m	.25 / .50 / 5.0	.25 / .50 / 5.0	.25 / .50 / 5.0	.25 / .50 / 5.0	.25 / .50 / 5.0	.25 / .50 / 5.0	.25 / .50 / 5.0	.25 / .50 / 5.0	.25 / .50 / 5.0
deg	2 / 5 / 10	2 / 5 / 10	2 / 5 / 10	2 / 5 / 10	2 / 5 / 10	2 / 5 / 10	2 / 5 / 10	2 / 5 / 10	2 / 5 / 10
AP-GEM [251]	1.4 / 14.2 / 65.9	9.6 / 29.4 / 82.9	2.4 / 19.1 / 80.5	3.6 / 20.3 / 78.1	4.4 / 21.5 / 86.0	4.5 / 15.8 / 75.9	1.8 / 7.5 / 58.2	0.0 / 0.2 / 6.8	0.1 / 1.2 / 15.8
DenseVLAD [127]	4.5 / 24.3 / 79.6	12.5 / 38.9 / 89.1	3.8 / 27.4 / 90.8	4.1 / 27.1 / 85.6	5.4 / 29.0 / 91.4	6.7 / 25.5 / 85.1	3.2 / 11.0 / 67.1	**1.4** / 2.7 / 23.2	0.6 / 5.2 / 29.8
NetVLAD [216]	2.2 / 16.8 / 73.3	11.4 / 31.0 / 85.9	3.2 / 21.5 / 90.9	4.1 / 22.6 / 84.0	4.2 / 22.2 / 89.4	5.2 / 20.1 / 80.8	2.4 / 10.4 / 70.3	0.2 / 1.2 / 9.1	0.3 / 0.9 / 8.8
DELG global [239]	1.6 / 10.9 / 66.4	8.9 / 23.9 / 81.3	2.1 / 16.5 / 77.6	3.5 / 18.5 / 73.6	3.9 / 20.5 / 87.9	3.6 / 13.5 / 73.5	1.0 / 6.4 / 59.6	0.2 / 0.7 / 7.6	0.1 / 1.6 / 13.8
DELG local [239]	1.7 / 10.4 / 78.3	2.5 / 7.3 / 76.8	1.1 / 8.9 / 84.2	1.2 / 9.1 / 83.2	1.2 / 4.5 / 76.8	3.5 / 10.9 / 80.8	3.3 / 12.6 / *85.2*	**1.4** / *7.6* / **38.6**	**2.4** / *11.9* / **53.0**
SuperGlue [238]	4.3 / 24.6 / 84.8	12.7 / 40.3 / 88.6	5.0 / 31.5 / *95.0*	*4.5* / 30.2 / 88.6	5.9 / 30.1 / 91.8	7.0 / 25.4 / 87.2	3.3 / 17.1 / 83.9	0.5 / 2.2 / 27.9	0.9 / 5.4 / *31.8*
Patch-NetVLAD [240]	4.8 / **72.5** / 86.2	**13.5** / **72.0** / 89.5	5.3 / **80.9** / 94.5	**6.3** / **71.3** / 89.8	5.9 / **79.3** / 92.1	7.8 / **75.9** / 87.9	4.8 / **67.3** / 83.4	0.5 / **12.4** / 24.9	*1.0* / **19.0** / 30.8
TransVPR [241]	**18.5** / 52.0 / **95.6**	10.7 / 44.7 / **100.0**	**12.3** / *45.5* / **99.1**	1.2 / *36.6* / **99.4**	**15.1** / 50.7 / **99.5**	**14.0** / *42.8* / **99.1**	**13.4** / *34.4* / **91.1**	*0.9* / 4.9 / *30.5*	0.0 / 1.0 / 10.3
CLASP-Net (Ours)	8.4 / 26.9 / *88.1*	5.1 / 25.9 / *89.8*	7.1 / 32.7 / 84.4	0.6 / 22.6 / *91.5*	*12.7* / 42.9 / *93.7*	*8.8* / 31.2 / *90.2*	*8.9* / 22.3 / 76.8	0.0 / 2.3 / 14.0	0.0 / 3.0 / 14.8

Table 9.4: Quantitative results on RobotCar Seasons v2 dataset. Best results are in **bold**. Second best results are in *italic*.

Tables (9.2), (9.3) and (9.4) show the results of CLASP-Net along with other approaches on the three previously described datasets: partitioned Nordland, Alderley Day/Night and RobotCar-Seasons datasets.

The results demonstrate that our method outperforms, by a large margin, standard baselines such as NetVLAD [216] and even local feature-based methods such as SuperGlue [238]. It outperforms Patch-NetVLAD [240] on Nordland dataset (Table (9.2)) and competes with it on Robotcar Seasons v2 (Table (9.4)), despite the fact that Patch-NetVLAD leverages multi-

scale descriptors whereas we rely on a single global descriptor. Only the transformer-based architecture TransVPR [241] presents a higher performance as compared to CLASP-Net. We note, however, that our model is based on simple ConvNet and MLP elements that can be upgraded to improve the performance. Finally, it is worth noting that we achieve state-of-the-art results on the very challenging Alderley dataset (Table (9.3)).

Qualitative results are presented in Figure (9.5) (Nordland dataset), Figure (9.6) (Alderley) and Figure (9.7) (Robotcar Seasons v2). More qualitative results are included in supplementary materials. One can see examples of queries and best retrieved images, along with Grad-CAM [252] activations. These visualizations demonstrate that CLASP-Net, even if trained without any labels, was able to learn features meaningful for outdoor localization tasks such as skylines for instance.

We focused our study on learning global visual representations that are robust to appearance changes and suitable for VPR. Our results demonstrate that it is possible to learn a model relying on constrastive self-supervision for robustness to appearance changes while being able to perceive the geometric structure of the input image by enforcing geometric prediction.

9.4.5 Discussion on Potential Limitations

Global image descriptors typically offer greater robustness to environmental conditions at the expense of being less tolerant to viewpoint changes compared to local descriptors [198]. Our approach aims to further enhance the robustness to environmental conditions, allowing it to handle extreme scenarios as seen in the Nordland or Alderley datasets effectively. However, it is important to acknowledge that our method may encounter limitations when dealing with datasets that feature significant viewpoint variations between reference and query images for the same location, as our slightly weaker performance on the Oxford RobotCar dataset suggests.

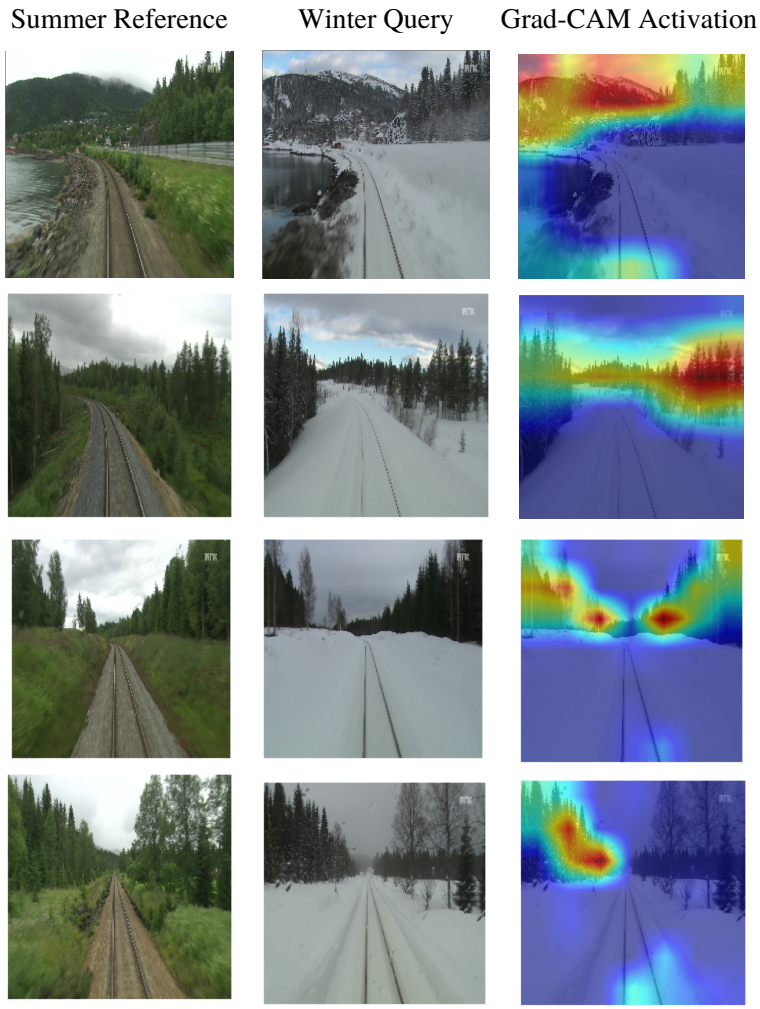


Figure 9.5: **Visual Grad-CAM activation of input query winter image**, along with retrieved summer image from the Nordland dataset.

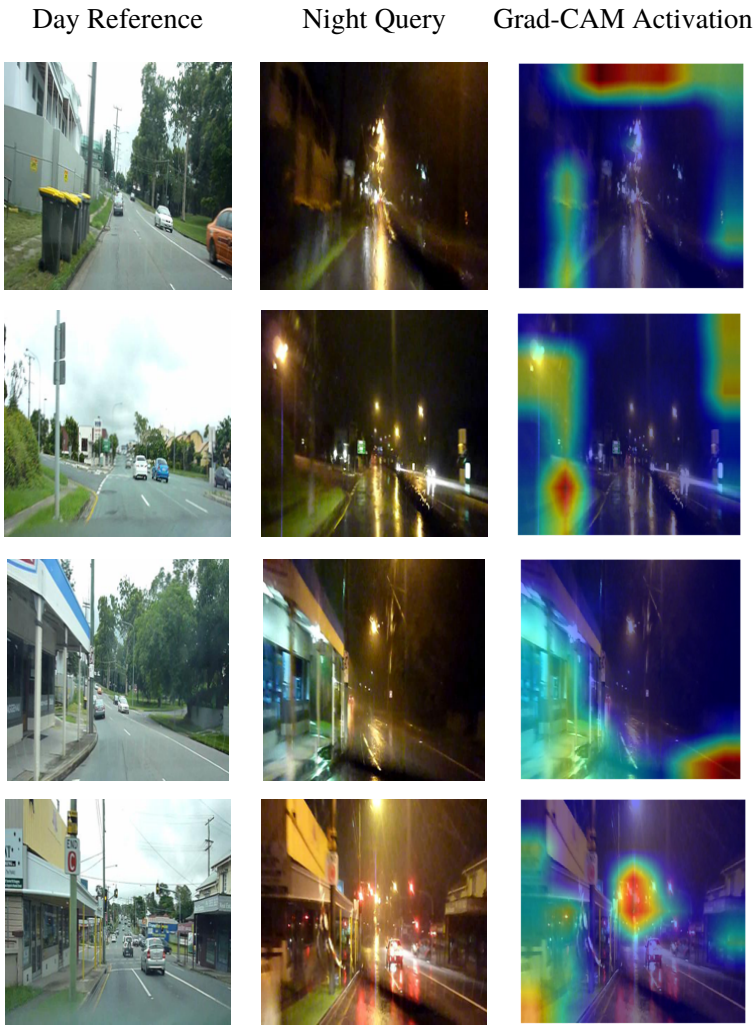


Figure 9.6: **Visual Grad-CAM activation of input query night image**, along with retrieved day image from the Alderley dataset.

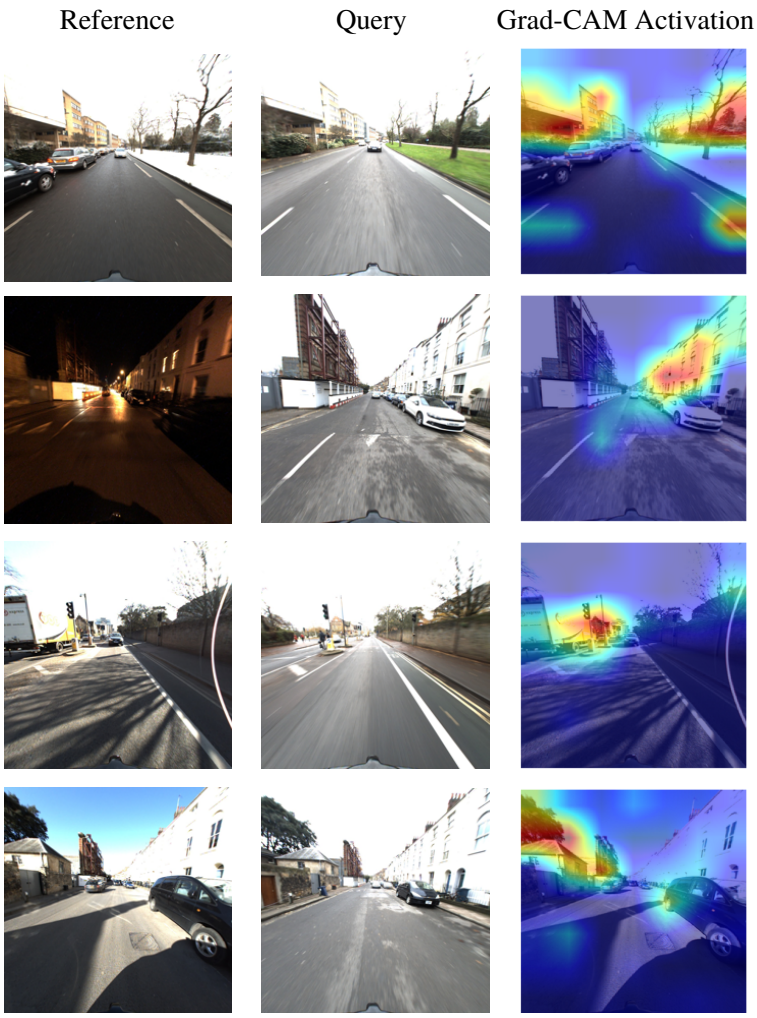


Figure 9.7: **Visual grad-CAM activation of input query image**, along with retrieved day image from the Oxford RobotCar Seasons v2 dataset.

9.5 Conclusions

In this chapter, we presented CLASP-Net, a novel self-supervised approach designed for visual place recognition under challenging appearance variations. A significant advantage of our method is its independence from human supervision. CLASP-Net is trained to learn features that are both robust to appearance changes and sensitive to geometric nuances, serving as abstract place representations useful for visual place recognition tasks. Our extensive experimental evaluations substantiate the effectiveness and efficiency of the proposed approach. As a direction for future research, we aim to extend our model's capabilities by exploring sensitivity to 3D geometric transformations through view synthesis techniques.

Chapter 10

Conclusions and Future Prospects

10.1 Summary and Highlights

The focus of this thesis centers on vision-based Space Situational Awareness (SSA) and the crucial task of estimating the relative pose of an uncooperative target in relation to a servicing spacecraft. This task is particularly vital during proximity operations such as fly-arounds, inspections, and close approaches in missions involving Active Debris Removal (ADR) and On-Orbit Servicing (OOS).

In our research, we thoroughly examined the intricacies and challenges associated with monocular vision-based systems for SSA. Such systems predominantly employ a single monocular camera as the primary navigation sensor. The preference for monocular cameras in these applications is attributed to their advantages in terms of reduced mass, lower power consumption, and simplified system complexity when compared to alternatives like active sensors or stereo camera systems. This thesis provides a comprehensive analysis of these systems, emphasizing their application and efficacy in the demanding realm of space navigation and situational awareness.

For training machine learning systems, reliable data generation methods are essential. Our work closely illustrates the SPARK simulation system we developed, which plays a critical role in generating synthetic images.

We detailed our work in the SPARK competition, an initiative specifically designed to

encourage research and innovation within the field of space target recognition and detection.

We explored the potential of an Equivariant Group Convolutional Neural Network (G-CNN)-based method for relative pose estimation, aiming to learn geometric features for direct pose estimation. This investigation was grounded in an understanding of the characteristics and limitations of existing methodologies.

Dissecting the system into its fundamental tasks is essential to effectively determine the capabilities and constraints of a relative pose estimation system. In particular, we argued that the principal limitations of monocular vision-based systems are often linked to their CNN-based feature extraction backbones. These backbones are tasked with extracting relevant information from 2D images.

While standard feature detectors in these systems have traditionally been optimized primarily for classification tasks, providing statistical insights about the presence of an object and its varying poses, they tend to need to improve in encoding sufficient spatial information in the extracted features. This limitation becomes particularly apparent in geometric inference tasks, such as relative pose estimation. Moreover, the performance of these systems can be significantly impaired in challenging scenarios, such as with symmetric objects or objects undergoing high degrees of transformation, such as in spacecraft pose estimation cases.

This thesis, therefore, focuses on addressing these shortcomings by enhancing the capability of CNNs to encode spatial and geometric information pertinent to relative pose estimation more effectively.

We also demonstrated the performance of our proposed equivariant pose estimation method tailored for spacecraft pose estimation. Our findings reveal that this method exhibits superior performance compared to other direct pose estimation approaches when tested on standard datasets for spacecraft pose estimation. This highlights the effectiveness and advanced capabilities of our approach in accurately determining the spacecraft pose, a critical aspect in the field of space situational awareness and related applications.

We introduced an innovative method for estimating the temporally consistent 3D trajectory of space objects from footage captured by a single RGB camera. This approach is pivotal in SSA, particularly for applications in ADR and OOS, where an accurate under-

standing of space objects' trajectories is crucial. Typically, relying on data from a single image perspective leads to temporally inconsistent 3D position estimations, a challenge our method effectively addresses.

Our space object trajectory estimation methodology is structured into two main phases for enhanced accuracy. Initially, a convolutional neural network pinpoints the 2D location of the space object. Then, these 2D coordinates are transformed into 3D space using a temporal convolutional neural network, ensuring consistent temporal coherence in the 3D trajectory estimations.

Our findings demonstrate that incorporating temporal information enhances the smoothness and accuracy of 3D trajectory estimations for space objects. This advancement marks a significant contribution to spatial tracking and analysis techniques, offering improved reliability and precision in critical space mission scenarios.

We presented the performance of our proposed method for 3D trajectory estimation, applying it to data from various sources, both real and synthetic. This exercise emphasized the significance of understanding and addressing the domain gap between synthetic and real images, which is crucial for effectively training and testing these models. Such insights are particularly vital in the field of Space Situational Awareness and its related applications, where accurate trajectory predictions are essential for successful mission execution and space object tracking.

We presented our investigation into visual place recognition, specifically leveraging self-supervised learning techniques to develop image features that are robust to appearance variations and sensitive to geometric changes. This approach has shown impressive results in visual place recognition under diverse seasonal and lighting conditions, achieved without relying on human-annotated labels.

The relevance of these appearance-resilient and geometry-aware features is particularly pronounced for OOS systems operating in the often unpredictable space environment. These systems require sophisticated visual recognition capabilities for effective navigation and operation, and our self-supervised learning method offers a substantial advancement in this area.

In contrast to traditional methods that depend on supervised learning and face challenges in generalizing to unique conditions, our self-supervised approach provides a more adaptable solution. By integrating contrastive and predictive learning paradigms, this approach yields robust and geometrically sensitive image features, as evidenced by our results on standard benchmarks. These outcomes demonstrate the competitiveness of our method in visual place recognition and highlight the potential of self-supervised learning in bolstering the capabilities of OOS systems within the challenging conditions of space.

10.2 Prospects for Future Research

In the context of monocular vision-based Space Situational Awareness, this thesis has identified several avenues for future research, each aiming to address the complexities inherent in various subsystems of this field, from feature detection in image processing to filtering estimates of the full relative state.

The proposed future research directions are as follows:

1. **Development of a Data Engine**: A key direction is the development of a data engine designed to generate, train, and validate machine learning models, incorporating both simulation and lab environments into the loop. This integrated approach would expedite the development and validation of models for diverse tasks in OOS, ensuring rapid and efficient model optimization.

2. **Sim2Real Domain Adaptation**: The critical need to understand and address the domain gap between synthetic and real images. This understanding is pivotal as it holds the potential to unlock significant advancements in the field. Recognizing and effectively bridging this gap can lead to more robust and accurate models, enhancing their applicability and performance in real-world scenarios, particularly in the dynamic and challenging environments characteristic of space missions. This aspect of research is especially crucial for improving the training and validation of machine learning models in On-Orbit Servicing (OOS) and other space-related applications, where the fidelity of

image data plays a key role in the success of autonomous operations and navigational tasks.

3. **Deep Learning Models with Inductive Biases**: Another promising area of research involves the utilization of DL models that leverage inductive biases—whether geometric or appearance-based—to create more compact models that nonetheless deliver superior performance. Such models could offer enhanced efficiency and accuracy, particularly beneficial in the context of space applications.

4. **Modular Design of Machine Learning and Classical Methods**: For spacecraft 3D trajectory estimation, a modular approach that combines machine learning techniques with classical methods presents a fruitful area for exploration. By integrating DL-based models with classical filtering techniques like Kalman or particle filters, we can achieve both accuracy and reliability. DL models, capable of handling large datasets, are great for identifying complex patterns but might struggle with variable conditions or require extensive computational resources. Classical methods offer real-time processing but can falter with intricate patterns or in the absence of accurate system dynamics. Combining these methods leverages the pattern recognition capabilities of DL models with the real-time processing strengths of classical methods, enhancing trajectory estimations for satellite tracking, collision avoidance, and navigation in complex orbital environments.

In conclusion, this thesis represents a comprehensive exploration into the realms of machine learning and image processing within the context of space situational awareness and On-Orbit Servicing. The work presented here lays a foundation for future innovations in the field, highlighting the immense potential of advanced computer vision and machine learning methods in addressing some of the most pressing challenges in space technology. As we look forward to the future, it is clear that the intersection of machine learning, image processing, and space technology holds exciting possibilities. The continued pursuit of understanding and bridging the domain gap between synthetic and real images, alongside the development of sophisticated algorithms and models, is set to profoundly influence the advancement of

space exploration and operation. This thesis, with its insights and findings, aims to contribute meaningfully to this ongoing journey of discovery and innovation in the vast expanse of space.

References

[1] Harry Jones. "The recent large reduction in space launch cost". In: 48th International Conference on Environmental Systems. 2018.

[2] Alexandra Witze. "2022 was a record year for space launches." In: *Nature* (2023).

[3] European Space Agency. *ESA Space Environment Report 2022*. https://www.sdo.esoc.esa.int/environment_report/Space_Environment_Report_latest.pdf. Accessed March 8, 2024. 2022.

[4] Joerg Kreisel. "On-Orbit servicing of satellites (OOS): its potential market & impact". In: *proceedings of 7th ESA Workshop on Advanced Space Technologies for Robotics and Automation 'ASTRA*. 2002.

[5] Minduli C Wijayatunga et al. "Design and guidance of a multi-active debris removal mission". In: *Astrodynamics* (2023), pp. 1–17.

[6] J Salvador Llorente et al. "PROBA-3: Precise formation flying demonstration mission". In: *Acta Astronautica* 82.1 (2013), pp. 38–46.

[7] Nola Taylor Redd. "Bringing satellites back from the dead: Mission extension vehicles give defunct spacecraft a new lease on life-[News]". In: *IEEE Spectrum* 57.8 (2020), pp. 6–7.

[8] Robin Biesbroek et al. "The clearspace-1 mission: ESA and clearspace team up to remove debris". In: *Proc. 8th Eur. Conf. Sp. Debris*. 2021, pp. 1–3.

[9] A. Yol et al. "Vision-based navigation in low earth orbit". In: *i-SAIRAS'16*. 2016. URL: https://hal.inria.fr/hal-01304728.

[10] Roberto Opromolla et al. “A review of cooperative and uncooperative spacecraft pose determination techniques for close-proximity operations”. In: *Progress in Aerospace Sciences* 93 (2017), pp. 53–72.

[11] Gabriella Gaias, Jean-Sébastien Ardaens, Simone D’Amico, et al. “The autonomous vision approach navigation and target identification (AVANTI) experiment: objectives and design”. In: *9th International ESA Conference on Guidance, Navigation & Control Systems, Porto, Portugal*. 2014.

[12] Guglielmo S Aglietti et al. “RemoveDEBRIS: An in-orbit demonstration of technologies for the removal of space debris”. In: *The Aeronautical Journal* 124.1271 (2020), pp. 1–23.

[13] American Aviation Publications. “Missiles & rockets”. English. In: *Missiles & rockets* 11.1-7 (Oct. 1957). Issues for Oct. 1957-May 1958 include section, Missile electronics, v. 11, no. 1-7. ”The weekly of advanced technology.”. ISSN: 0096-9702.

[14] Pablo Colmenarejo et al. “GNC aspects for active debris removal”. In: *CEAS EuroGNC. Delft, The Netherlands* (2013).

[15] J. L. Forshaw et al. “RemoveDEBRIS: An in-orbit active debris removal demonstration mission”. In: *Acta Astronautica* 127 (2016).

[16] Jill Davis and Henry Pernicka. “Proximity operations about and identification of non-cooperative resident space objects using stereo imaging”. In: *Acta Astronautica* 155 (2019), pp. 418–425.

[17] Vincenzo Pesce, Michèle Lavagna, and Riccardo Bevilacqua. “Stereovision-based pose and inertia estimation of unknown and uncooperative space objects”. In: *Advances in Space Research* 59.1 (2017), pp. 236–251.

[18] Roberto Opromolla et al. “Uncooperative pose estimation with a LIDAR-based system”. In: *Acta Astronautica* 110 (2015), pp. 287–297.

[19] Lorenzo Pasqualetto Cassinis, Robert Fonod, and Eberhard Gill. "Review of the robustness and applicability of monocular pose estimation systems for relative navigation with an uncooperative spacecraft". In: *Progress in Aerospace Sciences* 110 (2019), p. 100548.

[20] Lorenzo Pasqualetto Cassinis et al. "On-ground validation of a CNN-based monocular pose estimation system for uncooperative spacecraft: Bridging domain shift in rendezvous scenarios". In: *Acta Astronautica* 196 (2022), pp. 123–138.

[21] Jed M Kelsey et al. "Vision-based relative pose estimation for autonomous rendezvous and docking". In: *2006 IEEE aerospace conference*. IEEE. 2006, 20–pp.

[22] Simone D'Amico, Mathias Benn, and John L Jørgensen. "Pose estimation of an uncooperative spacecraft from actual space imagery". In: *International Journal of Space Science and Engineering 5* 2.2 (2014), pp. 171–189.

[23] Leo Pauly et al. "A survey on deep learning-based monocular spacecraft pose estimation: Current state, limitations and prospects". In: *Acta Astronautica* (2023).

[24] Derek Hoiem and Silvio Savarese. *Representations and techniques for 3D object recognition and scene interpretation*. Springer Nature, 2022.

[25] David G. Lowe. "Distinctive Image Features from Scale-Invariant Keypoints". In: *Int. J. Comput. Vis.* 60.2 (2004), pp. 91–110.

[26] William T. Freeman and Edward H. Adelson. "The Design and Use of Steerable Filters". In: *IEEE Trans. Pattern Anal. Mach. Intell.* 13.9 (1991), pp. 891–906.

[27] Jerry Jun Yokono and Tomaso A. Poggio. "Oriented Filters for Object Recognition: an Empirical Study". In: *Sixth IEEE International Conference on Automatic Face and Gesture Recognition (FGR 2004), May 17-19, 2004, Seoul, Korea*. 2004, pp. 755–760.

[28] Uwe Schmidt and Stefan Roth. "Learning rotation-aware features: From invariant priors to equivariant descriptors". In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2012.

[29] Mario Ferraro and Terry M Caelli. “Lie transformation groups, integral transforms, and invariant pattern recognition.” In: *Spatial Vision* (1994).

[30] Klas Nordberg and Gösta H. Granlund. “Equivariance and invariance-an approach based on Lie groups”. In: *Proceedings 1996 International Conference on Image Processing, Lausanne, Switzerland, September 16-19, 1996*. 1996, pp. 181–184.

[31] Osman Semih Kayhan and Jan C. van Gemert. “On Translation Invariance in CNNs: Convolutional Layers Can Exploit Absolute Spatial Location”. In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2020.

[32] Md Amirul Islam et al. “Position, padding and predictions: A deeper look at position information in cnns”. In: *arXiv preprint arXiv:2101.12322* (2021).

[33] Chris Olah et al. “Naturally Occurring Equivariance in Neural Networks”. In: *Distill* (2020). https://distill.pub/2020/circuits/equivariance. DOI: 10.23915/distill.00024.004.

[34] Joan Bruna and Stéphane Mallat. “Invariant Scattering Convolution Networks”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 35.8 (2013), pp. 1872–1886.

[35] Edouard Oyallon and Stephane Mallat. “Deep Roto-Translation Scattering for Object Classification”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2015.

[36] Erik J. Bekkers et al. “A Multi-Orientation Analysis Approach to Retinal Vessel Tracking”. In: *J. Math. Imaging Vis.* 49.3 (2014), pp. 583–610.

[37] Michiel Janssen et al. “Design and Processing of Invertible Orientation Scores of 3D Images”. In: *J. Math. Imaging Vis.* 60.9 (2018), pp. 1427–1458.

[38] Taco Cohen and Max Welling. “Group equivariant convolutional networks”. In: *International conference on machine learning*. PMLR. 2016, pp. 2990–2999.

[39] Carlos Henrique Machado Silva Esteves. “Learning Equivariant Representations”. PhD thesis. University of Pennsylvania, 2020.

[40] Soledad Villar et al. “Scalars are universal: Equivariant machine learning, structured like classical physics”. In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 28848–28863.

[41] Maurice Weiler and Gabriele Cesa. “General E(2)-Equivariant Steerable CNNs”. In: *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*. 2019, pp. 14334–14345.

[42] Marysia Winkels and Taco S Cohen. “3D G-CNNs for pulmonary nodule detection”. In: *arXiv preprint arXiv:1804.04656* (2018).

[43] Simon Batzner et al. “E (3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials”. In: *Nature communications* 13.1 (2022), p. 2453.

[44] Manuel Lopez-Antequera et al. “Appearance-invariant place recognition by discriminatively training a convolutional neural network”. In: *Pattern Recognition Letters* 92 (2017), pp. 89–95.

[45] Ruben Gomez-Ojeda et al. “Training a convolutional neural network for appearance-invariant place recognition”. In: *arXiv preprint arXiv:1505.07428* (2015).

[46] Junjun Wu et al. “Learning invariant semantic representation for long-term robust visual localization”. In: *Eng. Appl. Artif. Intell.* 111 (2022), p. 104793.

[47] Moritz Venator et al. “Self-Supervised Learning of Domain-Invariant Local Features for Robust Visual Localization Under Challenging Conditions”. In: *IEEE Robotics Autom. Lett.* 6.2 (2021), pp. 2753–2760.

[48] Jian Chen et al. “A partial intensity invariant feature descriptor for multimodal retinal image registration”. In: *IEEE Transactions on Biomedical Engineering* 57.7 (2010), pp. 1707–1718.

[49] Raia Hadsell, Sumit Chopra, and Yann LeCun. “Dimensionality reduction by learning an invariant mapping”. In: *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’06)*. Vol. 2. IEEE. 2006, pp. 1735–1742.

[50] Connor Shorten and Taghi M Khoshgoftaar. “A survey on image data augmentation for deep learning”. In: *Journal of big data* 6.1 (2019), pp. 1–48.

[51] Alessandro Achille and Stefano Soatto. “On the emergence of invariance and disentangling in deep representations”. In: *arXiv preprint arXiv:1706.01350* 125 (2017), pp. 126–127.

[52] Eric Marchand et al. “RemoveDebris vision-based navigation preliminary results”. In: *IAC 2019-70th International Astronautical Congress*. 2019, pp. 1–10.

[53] Maurice Weiler et al. *Equivariant and Coordinate Independent Convolutional Networks. A Gauge Field Theory of Neural Networks*. 2023. URL: https://maurice-weiler.gitlab.io/cnn_book/EquivariantAndCoordinateIndependentCNNs.pdf.

[54] Yann LeCun et al. “Gradient-based learning applied to document recognition”. In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324.

[55] Yann LeCun et al. “Handwritten Digit Recognition with a Back-Propagation Network”. In: *Advances in Neural Information Processing Systems 2, [NIPS Conference, Denver, Colorado, USA, November 27-30, 1989]*. 1989, pp. 396–404.

[56] Qianli Liao and Tomaso Poggio. *Exact equivariance, disentanglement and invariance of transformations*. Tech. rep. 2017.

[57] Laurent Sifre and Stéphane Mallat. “Rotation, scaling and deformation invariant scattering for texture discrimination”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2013, pp. 1233–1240.

[58] Angjoo Kanazawa, Abhishek Sharma, and David Jacobs. “Locally scale-invariant convolutional neural networks”. In: *arXiv preprint arXiv:1412.5104* (2014).

[59] Michael M Bronstein et al. “Geometric deep learning: Grids, groups, graphs, geodesics, and gauges”. In: *arXiv preprint arXiv:2104.13478* (2021).

[60] Taco S Cohen, Mario Geiger, and Maurice Weiler. “A general theory of equivariant cnns on homogeneous spaces”. In: *Advances in neural information processing systems* 32 (2019).

[61] Phillip Lippe. *UvA Deep Learning Tutorials*. https://uvadlc-notebooks.readthedocs.io/en/latest/. 2022.

[62] P. F Proença and Y. Gao. "Deep learning for spacecraft pose estimation from photo-realistic rendering". In: *2020 IEEE Int. Conf. on Robotics and Automation (ICRA)*. 2020.

[63] M. Kisantal et al. "Satellite Pose Estimation Challenge: Dataset,Competition Design and Results". In: *IEEE Trans. ON Aerospace AND Electronic Systems* (2020).

[64] Patrick W Kenneally, Scott Piggott, and Hanspeter Schaub. "Basilisk: A flexible, scalable and modular astrodynamics simulation framework". In: *Journal of aerospace information systems* 17.9 (2020), pp. 496–507.

[65] STK Ansys. *Software for Digital Mission Engineering and Systems Analysis*. 2022.

[66] a.i. Solutions. *FreeFlyer*. https://ai-solutions.com/freeflyer/. Software Package Version 7.8, Lanham, MD. 2018. URL: https://ai-solutions.com/freeflyer/ (visited on 10/21/2023).

[67] Iain Martin, Martin Dunstan, and Manuel Sanchez Gestido. "Planetary surface image generation for testing future space missions with pangu". In: *2nd RPI Space Imaging Workshop*. Sensing, Estimation, and Automation Laboratory. 2019.

[68] NASA Goddard Space Flight Center. *General Mission Analysis Tool*. R2018a. Version 17.4.0. Greenbelt, MD, 2018. URL: https://software.nasa.gov/software/GSC-17177-1.

[69] NASA Johnson Space Center. *Trick Simulation Environment*. Software. Houston, TX, 2023. URL: https://nasa.github.io/trick/ (visited on 10/20/2023).

[70] CS Systèmes d'Information. *OreKit Software Package Version 12.0: An Accurate and Efficient Core Layer for Space Flight Dynamics Applications*. https://www.orekit.org. Software Package Version 12.0. 2023.

[71] Roland Brochard et al. "Scientific image rendering for space scenes with the SurRender software". In: *arXiv preprint arXiv:1810.01423* (2018).

[72] Abhinandan Jain. "Darts-multibody modeling, simulation and analysis software". In: *Multibody Dynamics 2019: Proceedings of the 9th ECCOMAS Thematic Conference on Multibody Dynamics*. Springer. 2020, pp. 433–441.

[73] *Unity3D*. https://unity.com/.

[74] Unity Technologies. *Unity Perception Package*. https://github.com/Unity-Technologies/com.unity.perception. 2020.

[75] Leo Pauly et al. *Lessons from a Space Lab – An Image Acquisition Perspective*. 2022. DOI: 10.48550/ARXIV.2208.08865. URL: https://arxiv.org/abs/2208.08865.

[76] *NASA 3D Resources*. https://nasa3d.arc.nasa.gov/.

[77] Blender Online Community. *Blender - a 3D modelling and rendering package*. Blender Foundation. Stichting Blender Foundation, Amsterdam, 2018. URL: http://www.blender.org.

[78] M. Strube et al. "Raven: An on-orbit relative navigation demonstration using international space station visiting vehicles". In: *Advances in the Astronautical Sciences Guidance, Navigation and Control* 154 (2015).

[79] T. Chabot et al. "Vision-based navigation experiment onboard the removedebris mission". In: *10th Int. ESA Conference on GNC Systems*. 2017. URL: https://hal.inria.fr/hal-01784234.

[80] Sumant Sharma, Connor Beierle, and Simone D'Amico. "Pose estimation for non-cooperative spacecraft rendezvous using convolutional neural networks". In: *2018 IEEE Aerospace Conference*. IEEE. 2018, pp. 1–12.

[81] M. Kisantal et al. "Satellite Pose Estimation Challenge: Dataset,Competition Design and Results". In: *IEEE Trans. On Aerospace and Electronic Systems* (2020).

[82] *Pose Estimation Challenge*. https://kelvins.esa.int/satellite-pose-estimation-challenge/.

[83] Yinlin Hu et al. “Wide-Depth-Range 6D Object Pose Estimation in Space”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2021, pp. 15870–15879.

[84] Hoang Anh Dung, Bo Chen, and Tat-Jun Chin. “A Spacecraft Dataset for Detection, Segmentation and Parts Recognition”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*. June 2021, pp. 2012–2019.

[85] L. Liu et al. “Deep learning for generic object detection: A survey”. In: *Int. journal of computer vision* 128 (2020).

[86] K. He et al. “Deep Residual Learning for Image Recognition”. In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*. 2016.

[87] M. Tan, R. Pang, and Quoc V Le. “Efficientdet: Scalable and efficient object detection”. In: *IEEE/CVF conference on computer vision and pattern recognition*. 2020.

[88] Joseph Redmon et al. *You Only Look Once: Unified, Real-Time Object Detection*. 2016. arXiv: 1506.02640 [cs.CV].

[89] Torsten Sattler et al. “Understanding the Limitations of CNN-Based Absolute Camera Pose Regression”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2019.

[90] Liangchen Song et al. “Human pose estimation and its application to action recognition: A survey”. In: *Journal of Visual Communication and Image Representation* 76 (2021), p. 103055. ISSN: 1047-3203. DOI: https://doi.org/10.1016/j.jvcir.2021.103055. URL: https://www.sciencedirect.com/science/article/pii/S1047320321000262.

[91] Vincent Lepetit and Pascal Fua. “Monocular Model-Based 3D Tracking of Rigid Objects: A Survey”. In: *Found. Trends Comput. Graph. Vis.* 1.1 (2005).

[92] Éric Marchand, Hideaki Uchiyama, and Fabien Spindler. "Pose Estimation for Augmented Reality: A Hands-On Survey". In: *IEEE Trans. Vis. Comput. Graph.* 22.12 (2016), pp. 2633–2651.

[93] Martin A. Fischler and Robert C. Bolles. "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography". In: *Commun. ACM* 24.6 (1981), pp. 381–395.

[94] Vincent Lepetit, Francesc Moreno-Noguer, and Pascal Fua. "EPnP: An Accurate $O(n)$ Solution to the PnP Problem". In: *Int. J. Comput. Vis.* 81.2 (2009), pp. 155–166.

[95] Chi Xu et al. "Pose Estimation from Line Correspondences: A Complete Analysis and a Series of Solutions". In: *IEEE Trans. Pattern Anal. Mach. Intell.* 39.6 (2017), pp. 1209–1222.

[96] Kiru Park, Timothy Patten, and Markus Vincze. "Pix2Pose: Pixel-Wise Coordinate Regression of Objects for 6D Pose Estimation". In: *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*. IEEE, 2019, pp. 7667–7676.

[97] Mahbubul Alam et al. "Survey on Deep Neural Networks in Speech and Vision Systems". In: *Neurocomputing* 417 (2020), pp. 302–321.

[98] Taco S. Cohen and Max Welling. "Steerable CNNs". In: *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. 2017.

[99] Taco S. Cohen et al. "Spherical CNNs". In: *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. 2018.

[100] Haiwei Chen et al. "Equivariant Point Network for 3D Point Cloud Analysis". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2021, pp. 14514–14523.

[101] Jiaming Han et al. “ReDet: A Rotation-Equivariant Detector for Aerial Object Detection”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2021, pp. 2786–2795.

[102] Deepak K. Gupta, Devanshu Arya, and Efstratios Gavves. “Rotation Equivariant Siamese Networks for Tracking”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2021, pp. 12362–12371.

[103] Michael M. Bronstein et al. “Geometric Deep Learning: Grids, Groups, Graphs, Geodesics, and Gauges”. In: *CoRR* abs/2104.13478 (2021). arXiv: 2104.13478. URL: https://arxiv.org/abs/2104.13478.

[104] Carlos Esteves et al. “Cross-Domain 3D Equivariant Image Embeddings”. In: *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*. Ed. by Kamalika Chaudhuri and Ruslan Salakhutdinov. Vol. 97. Proceedings of Machine Learning Research. PMLR, 2019, pp. 1812–1822.

[105] Alex Kendall, Matthew Grimes, and Roberto Cipolla. “PoseNet: A Convolutional Network for Real-Time 6-DOF Camera Relocalization”. In: *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*. IEEE Computer Society, 2015, pp. 2938–2946.

[106] Christian Szegedy et al. “Going Deeper With Convolutions”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2015.

[107] Jian Wu, Liwei Ma, and Xiaolin Hu. “Delving deeper into convolutional neural networks for camera relocalization”. In: *2017 IEEE International Conference on Robotics and Automation (ICRA)*. 2017, pp. 5644–5651. DOI: 10.1109/ICRA.2017.7989663.

[108] Samarth Brahmbhatt et al. “Geometry-aware learning of maps for camera localization”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 2616–2625.

[109] Alex Kendall and Roberto Cipolla. “Modelling uncertainty in deep learning for camera relocalization”. In: *2016 IEEE International Conference on Robotics and Automation, ICRA 2016, Stockholm, Sweden, May 16-21, 2016*. 2016, pp. 4762–4769.

[110] Florian Walch et al. “Image-Based Localization Using LSTMs for Structured Feature Correlation”. In: *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*. 2017, pp. 627–637.

[111] Iaroslav Melekhov et al. “Image-Based Localization Using Hourglass Networks”. In: *2017 IEEE International Conference on Computer Vision Workshops, ICCV Workshops 2017, Venice, Italy, October 22-29, 2017*. IEEE Computer Society, 2017, pp. 870–877.

[112] Yinlin Hu et al. “Segmentation-Driven 6D Object Pose Estimation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2019.

[113] Tomas Hodan, Daniel Barath, and Jiri Matas. “EPOS: Estimating 6D Pose of Objects With Symmetries”. In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2020.

[114] Tayyab Naseer and Wolfram Burgard. “Deep regression for monocular camera-based 6-DoF global localization in outdoor environments”. In: *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2017, Vancouver, BC, Canada, September 24-28, 2017*. IEEE, 2017, pp. 1525–1530.

[115] Alex Kendall and Roberto Cipolla. “Geometric Loss Functions for Camera Pose Regression with Deep Learning”. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*. IEEE Computer Society, 2017, pp. 6555–6564.

[116] Ming Cai, Chunhua Shen, and Ian Reid. “A Hybrid Probabilistic Model for Camera Relocalization”. In: *British Machine Vision Conference 2018, BMVC 2018, Newcastle, UK, September 3-6, 2018*. 2018, p. 238.

[117] Yu Xiang et al. “PoseCNN: A Convolutional Neural Network for 6D Object Pose Estimation in Cluttered Scenes”. In: *Robotics: Science and Systems XIV, Carnegie Mellon University, Pittsburgh, Pennsylvania, USA, June 26-30, 2018*. 2018.

[118] Gu Wang et al. “GDR-Net: Geometry-Guided Direct Regression Network for Monocular 6D Object Pose Estimation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2021, pp. 16611–16621.

[119] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. `http://www.deeplearningbook.org`. MIT Press, 2016.

[120] Carlos Esteves et al. “Cross-domain 3d equivariant image embeddings”. In: *International Conference on Machine Learning*. PMLR. 2019, pp. 1812–1822.

[121] Chao Zhang et al. “Rotation Equivariant Orientation Estimation for Omnidirectional Localization”. In: *Proceedings of the Asian Conference on Computer Vision (ACCV)*. Nov. 2020.

[122] Kenichi Kanatani. *Group Theoretical Methods in Image Understanding*. Berlin, Heidelberg: Springer-Verlag, 1990. ISBN: 0387512535.

[123] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Second. Cambridge University Press, ISBN: 0521540518, 2004.

[124] Taco S. Cohen, Mario Geiger, and Maurice Weiler. “A General Theory of Equivariant CNNs on Homogeneous Spaces”. In: *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*. 2019, pp. 9142–9153.

[125] K. He et al. “Deep Residual Learning for Image Recognition”. In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*. 2016.

[126] Tomáš Hodaň et al. “T-LESS: An RGB-D Dataset for 6D Pose Estimation of Texture-less Objects”. In: *IEEE Winter Conference on Applications of Computer Vision (WACV)* (2017).

[127] Akihiko Torii et al. “24/7 place recognition by view synthesis”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 1808–1817.

[128] Alex Kendall and Roberto Cipolla. “Modelling uncertainty in deep learning for camera relocalization”. In: *2016 IEEE international conference on Robotics and Automation (ICRA)*. IEEE. 2016, pp. 4762–4769.

[129] Florian Walch et al. “Image-based localization using lstms for structured feature correlation”. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2017, pp. 627–637.

[130] Tayyab Naseer and Wolfram Burgard. “Deep regression for monocular camera-based 6-dof global localization in outdoor environments”. In: *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2017, pp. 1525–1530.

[131] Ming Cai, Chunhua Shen, and Ian Reid. “A hybrid probabilistic model for camera relocalization”. In: (2019).

[132] Yoli Shavit and Ron Ferens. “Do We Really Need Scene-specific Pose Encoders?” In: *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE. 2021, pp. 3186–3192.

[133] Shavit et al. “Learning Multi-Scene Absolute Pose Regression With Transformers”. In: *ICCV 2021*.

[134] Shavit et al. “Paying Attention to Activation Maps in Camera Pose Regression”. In: *CoRR* abs/2103.11477 (2021). URL: https://arxiv.org/abs/2103.11477.

[135] Bing Wang et al. “AtLoc: Attention Guided Camera Localization”. In: *arXiv preprint arXiv:1909.03557* (2019).

[136] Jamie Shotton et al. “Scene Coordinate Regression Forests for Camera Relocalization in RGB-D Images”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2013.

[137] Diederik P. Kingma and Jimmy Ba. “Adam: A Method for Stochastic Optimization”. In: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. Ed. by Yoshua Bengio and Yann LeCun. 2015.

[138] Mate Kisantal et al. “Satellite pose estimation challenge: Dataset, competition design, and results”. In: *IEEE Transactions on Aerospace and Electronic Systems* 56.5 (2020), pp. 4083–4098.

[139] Tae Ha Park et al. “Satellite pose estimation competition 2021: Results and analyses”. In: *Acta Astronautica* 204 (2023), pp. 640–665.

[140] Mate Kisantal et al. “Spacecraft pose estimation dataset (speed)”. In: *Zenodo, February* (2019).

[141] Tae Ha Park et al. “SPEED+: Next-generation dataset for spacecraft pose estimation across domain gap”. In: *2022 IEEE Aerospace Conference (AERO)*. IEEE. 2022, pp. 1–15.

[142] Bo Chen et al. “Satellite pose estimation with deep landmark regression and nonlinear pose refinement”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*. 2019.

[143] Shaoqing Ren et al. “Faster r-cnn: Towards real-time object detection with region proposal networks”. In: *Advances in neural information processing systems* 28 (2015).

[144] Jingdong Wang et al. “Deep high-resolution representation learning for visual recognition”. In: *IEEE transactions on pattern analysis and machine intelligence* 43.10 (2020), pp. 3349–3364.

[145] Kyle Gerard. “Segmentation-driven satellite pose estimation”. In: *Kelvins Day Presentation, URL: https://indico. esa. int/event/319/attachments/3561/4754/pose_ gerard_segmentation. pdf* (2019).

[146] Tae Ha Park, Sumant Sharma, and Simone D'Amico. "Towards robust learning-based pose estimation of noncooperative spacecraft". In: *arXiv preprint arXiv:1909.00392* (2019).

[147] Joseph Redmon and Ali Farhadi. "Yolov3: An incremental improvement". In: *arXiv preprint arXiv:1804.02767* (2018).

[148] Mark Sandler et al. "Mobilenetv2: Inverted residuals and linear bottlenecks". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 4510–4520.

[149] Vincent Gaudillière et al. "3D-Aware Object Localization using Gaussian Implicit Occupancy Function". In: *IROS 2023 – 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Detroit, United States, Oct. 2023.

[150] Zi Wang et al. "Revisiting monocular satellite pose estimation with transformer". In: *IEEE Transactions on Aerospace and Electronic Systems* 58.5 (2022), pp. 4279–4294.

[151] Kaiming He et al. "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[152] M. Fischler and R. Bolles. "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography". In: *Communications of the ACM* 24.6 (1981), pp. 381–395. URL: /brokenurl#%20http://publication.wilsonwong.me/load.php?id=233282275.

[153] Vincent Lepetit, Francesc Moreno-Noguer, and P Fua. "EPnP: Efficient perspective-n-point camera pose estimation". In: *Int. J. Comput. Vis* 81.2 (2009), pp. 155–166.

[154] Stephen J Wright. *Numerical optimization*. 2006.

[155] Thaweerath Phisannupawong et al. "Vision-based spacecraft pose estimation via a deep convolutional neural network for noncooperative docking operations". In: *Aerospace* 7.9 (2020), p. 126.

[156] Sumant Sharma and Simone D'Amico. "Pose estimation for non-cooperative rendezvous using neural networks". In: *arXiv preprint arXiv:1906.09868* (2019).

[157] Julien Posso, Guy Bois, and Yvon Savaria. "Mobile-URSONet: an Embeddable Neural Network for Onboard Spacecraft Pose Estimation". In: *2022 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE. 2022, pp. 794–798.

[158] C. Priyant Mark and Surekha Kamath. "Review of Active Space Debris Removal Methods". In: *Space Policy* 47 (2019), pp. 194–206. ISSN: 0265-9646. DOI: https://doi.org/10.1016/j.spacepol.2018.12.005. URL: https://www.sciencedirect.com/science/article/pii/S0265964618300110.

[159] Bruno Esmiller and Christophe Jacquelard. "CLEANSPACE "Small Debris Removal By Laser Illumination And Complementary Technologies"". In: *AIP Conference Proceedings* 1402 (Nov. 2011), pp. 347–353. DOI: 10.1063/1.3657041.

[160] DE Olmos et al. "ANDROID small active debris removal mission". In: *Proceedings of the Fifth CEAS Air and Space conference, Delft, Netherlands*. 2015, pp. 7–11.

[161] Robin Biesbroek et al. "THE CLEARSPACE-1 MISSION: ESA AND CLEARSPACE TEAM UP TO REMOVE DEBRIS". In: ().

[162] J. Martinez et al. "A simple yet effective baseline for 3d human pose estimation". In: *ICCV*. 2017.

[163] Dario Pavllo et al. "3D human pose estimation in video with temporal convolutions and semi-supervised training". In: *arXiv preprint arXiv:1811.11742* (2018).

[164] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. *U-Net: Convolutional Networks for Biomedical Image Segmentation*. 2015. arXiv: 1505.04597 [cs.CV].

[165] Aiden Nibali et al. *Numerical Coordinate Regression with Convolutional Neural Networks*. 2018. arXiv: 1801.07372 [cs.CV].

[166] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. *Objects as Points*. 2019. arXiv: 1904.07850 [cs.CV].

[167] Mohamed Adel Musallam et al. "Temporal 3d human pose estimation for action recognition from arbitrary viewpoints". In: *2019 International Conference on Computational Science and Computational Intelligence (CSCI)*. IEEE. 2019, pp. 253–258.

[168] Nikhil Mishra et al. *A Simple Neural Attentive Meta-Learner*. 2017. arXiv: 1707.03141 [cs.AI].

[169] Matthias Holschneider et al. "A real-time algorithm for signal analysis with the help of the wavelet transform". In: *Wavelets*. Springer, 1990, pp. 286–297.

[170] S. Bai, J. Z. Kolter, and V. Koltun. "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling". In: *arXiv:1803.01271* (2018).

[171] Aaron van den Oord et al. "Wavenet: A generative model for raw audio". In: *arXiv preprint arXiv:1609.03499* (2016).

[172] Fisher Yu and Vladlen Koltun. "Multi-scale context aggregation by dilated convolutions". In: *arXiv preprint arXiv:1511.07122* (2015).

[173] Nal Kalchbrenner et al. "Neural machine translation in linear time". In: *arXiv preprint arXiv:1610.10099* (2016).

[174] Sepp Hochreiter and Jürgen Schmidhuber. "Long Short-Term Memory". In: *Neural Comput.* 9.8 (Nov. 1997), pp. 1735–1780. ISSN: 0899-7667. DOI: 10.1162/neco.1997.9.8.1735. URL: http://dx.doi.org/10.1162/neco.1997.9.8.1735.

[175] KyungHyun Cho et al. "On the Properties of Neural Machine Translation: Encoder-Decoder Approaches". In: *CoRR* abs/1409.1259 (2014). arXiv: 1409.1259. URL: http://arxiv.org/abs/1409.1259.

[176] Albert Garcia et al. "LSPnet: A 2D Localization-oriented Spacecraft Pose Estimation Neural Network". In: *CoRR* abs/2104.09248 (2021). arXiv: 2104.09248. URL: https://arxiv.org/abs/2104.09248.

[177] GSFC. *NASA's Exploration & In-space Services*. en. publisher: NASA. URL: https://nexis.gsfc.nasa.gov/ (visited on 02/14/2022).

[178] A Pellacani et al. "HERA vision based GNC and autonomy". In: *8 TH EUROPEAN CONFERENCE FOR AERONAUTICS AND SP* (2019). DOI: 10.13009/EUCASS2019-39.

[179] Mate Kisantal et al. “Satellite pose estimation challenge: Dataset, competition design, and results”. In: *IEEE Transactions on Aerospace and Electronic Systems* 56.5 (2020), pp. 4083–4098.

[180] Mohamed Adel Musallam et al. “Spacecraft Recognition Leveraging Knowledge of Space Environment: Simulator, Dataset, Competition Design and Analysis”. In: *2021 IEEE International Conference on Image Processing Challenges (ICIPC)*. IEEE. 2021, pp. 11–15.

[181] Tae Ha Park et al. “SPEED+: Next Generation Dataset for Spacecraft Pose Estimation across Domain Gap”. In: *arXiv preprint arXiv:2110.03101* (2021).

[182] Pedro F Proença and Yang Gao. “Deep learning for spacecraft pose estimation from photorealistic rendering”. In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2020, pp. 6007–6013.

[183] Yinlin Hu et al. “Wide-Depth-Range 6D Object Pose Estimation in Space”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 15870–15879.

[184] Kevin Black et al. “Real-time, flight-ready, non-cooperative spacecraft pose estimation using monocular imagery”. In: *arXiv preprint arXiv:2101.09553* (2021).

[185] Tae Ha Park, Sumant Sharma, and Simone D’Amico. “Towards robust learning-based pose estimation of noncooperative spacecraft”. In: *arXiv preprint arXiv:1909.00392* (2019).

[186] Simone D’Amico et al. “Prisma”. In: *Distributed Space Missions for Earth System Monitoring*. Springer, 2013, pp. 599–637.

[187] *Blender 3.0 Reference Manual — Blender Manual*. URL: https://docs.blender.org/manual/en/latest/index.html.

[188] Stanford University. *Space Rendezvous Laboratory*. Accessed on 2022-03-11. URL: https://damicos.people.stanford.edu/.

[189] Takafumi Taketomi, Hideaki Uchiyama, and Sei Ikeda. “Visual SLAM algorithms: a survey from 2010 to 2016”. In: *IPSJ Trans. Comput. Vis. Appl.* 9 (2017), p. 16. DOI: 10.1186/s41074-017-0027-2.

[190] Enric Corona, Kaustav Kundu, and Sanja Fidler. “Pose Estimation for Objects with Rotational Symmetry”. In: *IEEE International Conference on Intelligent Robots and Systems* (Dec. 2018), pp. 7215–7222. ISSN: 21530866. DOI: 10.1109/IROS.2018.8594282.

[191] Giorgia Pitteri et al. “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*. IEEE, 2019, pp. 614–622.

[192] Jiaming Hu et al. “Pose Estimation of Specular and Symmetrical Objects”. In: *CoRR* abs/2011.00372 (2020).

[193] Mohamed Adel Musallam et al. “Leveraging Temporal Information for 3D Trajectory Estimation of Space Objects”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*. Oct. 2021, pp. 3816–3822.

[194] Mingxing Tan and Quoc V. Le. “EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks”. In: *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*. Ed. by Kamalika Chaudhuri and Ruslan Salakhutdinov. Vol. 97. Proceedings of Machine Learning Research. PMLR, 2019, pp. 6105–6114.

[195] Alex Kendall, Matthew Grimes, and Roberto Cipolla. “Posenet: A convolutional network for real-time 6-dof camera relocalization”. In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 2938–2946.

[196] Noé Pion et al. “Benchmarking Image Retrieval for Visual Localization”. In: *8th International Conference on 3D Vision, 3DV 2020, Virtual Event, Japan, November 25-28, 2020*. Ed. by Vitomir Struc and Francisco Gómez Fernández. IEEE, 2020, pp. 483–494.

[197] Martin Humenberger et al. “Investigating the Role of Image Retrieval for Visual Localization”. In: *Int. J. Comput. Vis.* 130.7 (2022), pp. 1811–1836.

[198] Stephanie Lowry et al. “Visual place recognition: A survey”. In: *IEEE Transactions on Robotics* 32.1 (2015), pp. 1–19.

[199] James C. R. Whittington et al. “How to build a cognitive map”. In: *Nature Neuroscience* 25.10 (Oct. 2022), pp. 1257–1272. ISSN: 1546-1726. DOI: 10.1038/s41593-022-01153-y. URL: https://doi.org/10.1038/s41593-022-01153-y.

[200] Sourav Garg, Tobias Fischer, and Michael Milford. “Where Is Your Place, Visual Place Recognition?” In: *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021*. Ed. by Zhi-Hua Zhou. ijcai.org, 2021, pp. 4416–4425.

[201] Stephanie Lowry and Henrik Andreasson. “Lightweight, Viewpoint-Invariant Visual Place Recognition in Changing Environments”. In: *IEEE Robotics and Automation Letters* 3.2 (2018), pp. 957–964. DOI: 10.1109/LRA.2018.2793308.

[202] Yifei Wang et al. “Residual relaxation for multi-view representation learning”. In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 12104–12115.

[203] Rumen Dangovski et al. “Equivariant Self-Supervised Learning: Encouraging Equivariance in Representations”. In: *International Conference on Learning Representations*. 2022. URL: https://openreview.net/forum?id=gKLAAfiytI.

[204] Mohamed Adel Musallam et al. “Leveraging equivariant features for absolute pose regression”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 6876–6886.

[205] Robert Geirhos et al. “Shortcut learning in deep neural networks”. In: *Nature Machine Intelligence* 2.11 (Nov. 2020), pp. 665–673. ISSN: 2522-5839. DOI: 10.1038/s42256-020-00257-z. URL: https://doi.org/10.1038/s42256-020-00257-z.

[206] Yann LeCun and Ishan Misra. *Self-supervised learning: The dark matter of intelligence*. https://ai.facebook.com/blog/self-supervised-learning-the-dark-matter-of-intelligence/. Consulted: October, 2022. Mar. 2021.

[207] Sourav Garg, Madhu Babu Vankadari, and Michael Milford. "SeqMatchNet: Contrastive Learning with Sequence Matching for Place Recognition & Relocalization". In: *Conference on Robot Learning, 8-11 November 2021, London, UK*. Ed. by Aleksandra Faust, David Hsu, and Gerhard Neumann. Vol. 164. Proceedings of Machine Learning Research. PMLR, 2021, pp. 429–443.

[208] Li Tang et al. "Explicit feature disentanglement for visual place recognition across appearance changes". In: *International Journal of Advanced Robotic Systems* 18.6 (2021), p. 17298814211037497. DOI: 10.1177/17298814211037497. eprint: https://doi.org/10.1177/17298814211037497. URL: https://doi.org/10.1177/17298814211037497.

[209] Ting Chen et al. "A Simple Framework for Contrastive Learning of Visual Representations". In: *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*. Vol. 119. Proceedings of Machine Learning Research. PMLR, 2020, pp. 1597–1607.

[210] Longlong Jing and Yingli Tian. "Self-Supervised Visual Feature Learning With Deep Neural Networks: A Survey". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 43.11 (2021), pp. 4037–4058. DOI: 10.1109/TPAMI.2020.2992393.

[211] Michael J Milford and Gordon F Wyeth. "SeqSLAM: Visual route-based navigation for sunny summer days and stormy winter nights". In: *2012 IEEE international conference on robotics and automation*. IEEE. 2012, pp. 1643–1649.

[212] Zetao Chen et al. "Convolutional neural network-based place recognition". In: *arXiv preprint arXiv:1411.1509* (2014).

[213] Niko Sünderhauf et al. "On the performance of convnet features for place recognition". In: *2015 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE. 2015, pp. 4297–4304.

[214] Yi Hou, Hong Zhang, and Shilin Zhou. "Convolutional neural network-based image representation for visual loop closure detection". In: *2015 IEEE international conference on information and automation*. IEEE. 2015, pp. 2238–2245.

[215] Nate Merrill and Guoquan Huang. “Lightweight unsupervised deep loop closure”. In: *arXiv preprint arXiv:1805.07703* (2018).

[216] Relja Arandjelovic et al. “NetVLAD: CNN architecture for weakly supervised place recognition”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 5297–5307.

[217] Eng-Jon Ong et al. “Deep architectures and ensembles for semantic video classification”. In: *IEEE Transactions on Circuits and Systems for Video Technology* 29.12 (2018), pp. 3568–3582.

[218] Antoine Miech, Ivan Laptev, and Josef Sivic. “Learnable pooling with context gating for video classification”. In: *arXiv preprint arXiv:1706.06905* (2017).

[219] Richard Zhang, Phillip Isola, and Alexei A. Efros. “Colorful Image Colorization”. In: *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III*. Ed. by Bastian Leibe et al. Vol. 9907. Lecture Notes in Computer Science. Springer, 2016, pp. 649–666.

[220] Mehdi Noroozi and Paolo Favaro. “Unsupervised Learning of Visual Representations by Solving Jigsaw Puzzles”. In: *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VI*. Ed. by Bastian Leibe et al. Vol. 9910. Lecture Notes in Computer Science. Springer, 2016, pp. 69–84.

[221] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. “Unsupervised Representation Learning by Predicting Image Rotations”. In: *International Conference on Learning Representations*. 2018. URL: https://openreview.net/forum?id=S1v4N2l0-.

[222] Philip Bachman, R Devon Hjelm, and William Buchwalter. “Learning Representations by Maximizing Mutual Information Across Views”. In: *Advances in Neural Information Processing Systems*. Ed. by H. Wallach et al. Vol. 32. Curran Associates, Inc., 2019. URL: https://proceedings.neurips.cc/paper/2019/file/ddf354219aac374f1d40b7e760ee5bb7-Paper.pdf.

[223] Zhirong Wu et al. “Unsupervised Feature Learning via Non-Parametric Instance Discrimination”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2018.

[224] Ishan Misra and Laurens van der Maaten. “Self-Supervised Learning of Pretext-Invariant Representations”. In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2020.

[225] Vitaliy Kinakh, Olga Taran, and Svyatoslav Voloshynovskiy. “ScatSimCLR: Self-Supervised Contrastive Learning With Pretext Task Regularization for Small-Scale Datasets”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*. Oct. 2021, pp. 1098–1106.

[226] Mandela Patrick et al. “On Compositions of Transformations in Contrastive Self-Supervised Learning”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. Oct. 2021, pp. 9577–9587.

[227] Tan Wang et al. “Equivariance and Invariance Inductive Bias for Learning from Insufficient Data”. In: *Computer Vision - ECCV 2022 - 17th European Conference, Tel Aviv, Israel, October 23-27, 2022, Proceedings*. Springer, 2022.

[228] Quentin Garrido, Laurent Najman, and Yann Lecun. “Self-supervised learning of Split Invariant Equivariant representations”. In: *arXiv preprint arXiv:2302.10283* (2023).

[229] Robin Winter et al. “Unsupervised Learning of Group Invariant and Equivariant Representations”. In: *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, November 28-December 9, 2022, hybrid*. 2022.

[230] Zeyu Feng, Chang Xu, and Dacheng Tao. “Self-Supervised Representation Learning by Rotation Feature Decoupling”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2019.

[231] Niluthpol Chowdhury Mithun et al. “Learning Long-Term Invariant Features for Vision-Based Localization”. In: *2018 IEEE Winter Conference on Applications of Computer*

Vision, WACV 2018, Lake Tahoe, NV, USA, March 12-15, 2018. IEEE Computer Society, 2018, pp. 2038–2047.

[232] Janine Thoma, Danda Pani Paudel, and Luc V Gool. "Soft Contrastive Learning for Visual Localization". In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle et al. Vol. 33. Curran Associates, Inc., 2020, pp. 11119–11130. URL: https://proceedings.neurips.cc/paper/2020/file/7f2cba89a7116c7c6b0a769572d5fad9-Paper.pdf.

[233] Niko Sünderhauf, Peer Neubert, and Peter Protzel. "Are we there yet? Challenging SeqSLAM on a 3000 km journey across all four seasons". In: *Proc. of workshop on long-term autonomy, IEEE international conference on robotics and automation (ICRA)*. 2013, p. 2013.

[234] Daniel Olid, José M. Fácil, and Javier Civera. "Single-View Place Recognition under Seasonal Changes". In: *PPNIV Workshop at IROS 2018*. 2018.

[235] Carl Toft et al. "Long-term visual localization revisited". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2020).

[236] Will Maddern et al. "1 Year, 1000km: The Oxford RobotCar Dataset". In: *The International Journal of Robotics Research (IJRR)* 36.1 (2017), pp. 3–15.

[237] Yixiao Ge et al. "Self-supervising fine-grained region similarities for large-scale image localization". In: *European conference on computer vision*. Springer. 2020, pp. 369–386.

[238] Paul-Edouard Sarlin et al. "Superglue: Learning feature matching with graph neural networks". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 4938–4947.

[239] Bingyi Cao, Andre Araujo, and Jack Sim. "Unifying deep local and global features for image search". In: *European Conference on Computer Vision*. Springer. 2020, pp. 726–743.

[240] Stephen Hausler et al. “Patch-NetVLAD: Multi-Scale Fusion of Locally-Global Descriptors for Place Recognition”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2021, pp. 14141–14152.

[241] Ruotong Wang et al. “TransVPR: Transformer-Based Place Recognition With Multi-Level Attention Aggregation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2022, pp. 13648–13657.

[242] Jose M Facil et al. “Condition-invariant multi-view place recognition”. In: *arXiv preprint arXiv:1902.09516* (2019).

[243] Li Tang et al. “Adversarial feature disentanglement for place recognition across changing appearance”. In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2020, pp. 1301–1307.

[244] Luis G Camara and Libor Přeučil. “Visual place recognition by spatial matching of high-level CNN features”. In: *Robotics and Autonomous Systems* 133 (2020), p. 103625.

[245] Stephen Hausler and Michael Milford. “Hierarchical multi-process fusion for visual place recognition”. In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2020, pp. 3327–3333.

[246] Shuxue Peng et al. “MegLoc: A Robust and Accurate Visual Localization Pipeline”. In: *CoRR* abs/2111.13063 (2021).

[247] Ross Wightman. *PyTorch Image Models*. https://github.com/rwightman/pytorch-image-models. 2019. DOI: 10.5281/zenodo.4414861.

[248] E. Riba et al. “Kornia: an Open Source Differentiable Computer Vision Library for PyTorch”. In: *Winter Conference on Applications of Computer Vision*. 2020. URL: https://arxiv.org/pdf/1910.02190.pdf.

[249] Diederik P Kingma and Jimmy Ba. “Adam: A method for stochastic optimization”. In: *arXiv preprint arXiv:1412.6980* (2014).

[250] Ting Chen et al. “A simple framework for contrastive learning of visual representations”. In: *International conference on machine learning*. PMLR. 2020, pp. 1597–1607.

[251] Jerome Revaud et al. “Learning with average precision: Training image retrieval with a listwise loss”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, pp. 5107–5116.

[252] Ramprasaath R. Selvaraju et al. “Grad-CAM: Visual Explanations From Deep Networks via Gradient-Based Localization”. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. Oct. 2017.