Degree Project in Technology

Second cycle, 30 credits

Multi-Modal Place Recognition and Pose Estimation for Autonomous Rovers in Unstructured Environments

From Image Retrieval to 6D Pose Estimation for Loop Closure in SLAM

LAURA ALEJANDRA ENCINAR GONZALEZ

Multi-Modal Place Recognition and Pose Estimation for Autonomous Rovers in Unstructured Environments

From Image Retrieval to 6D Pose Estimation for Loop Closure in SLAM

LAURA ALEJANDRA ENCINAR GONZALEZ

Master's Programme, ICT Innovation, 120 credits
Date: August 22, 2025

Supervisors: John Folkesson, Riccardo Giubilato, Quan Zhou
Examiner: Patric Jensfelt
School of Electrical Engineering and Computer Science
Host organization: German Aerospace Center (DLR), Institute of Robotics and Mechatronics
Swedish title: Multi-Modal Platsigenkänning och Positionsuppskattning för Autonoma Rovers i Ostrukturerade Miljöer
Swedish subtitle: Från bildtagning till 6D lägesbestämning för loop-stängning i SLAM

Abstract

Autonomous navigation in planetary-like environments presents unique challenges due to the absence of GPS signals, limited semantic structure, and visual ambiguity caused by repetitive textures or harsh lighting conditions. Traditional place recognition and localization methods either rely on dense maps and structured environments or only provide coarse retrieval without estimating full 6-DoF (Degrees of Freedom) poses. This limits their applicability in the context of real-time Simultaneous Localization and Mapping (SLAM) for field robotics and planetary exploration.

This thesis addresses the problem by developing a multi-modal system that performs both place recognition and relative pose estimation in unstructured, GNSS-denied environments. The proposed approach fuses visual features extracted from a transformer-based encoder (DINOv2) with 3D geometric descriptors from a LiDAR-based backbone (SONATA). These features are projected and aligned in 3D space to produce interpretable correspondences, from which the system estimates full 6D poses. On the retrieval side, DINOv2 descriptors are aggregated using SALAD, a learned VLAD-style module, and searched efficiently using FAISS indexing. The system is evaluated on the Etna volcano dataset, representative of planetary terrains.

The results show that the proposed model outperforms established retrieval methods like NetVLAD and TransVPR and achieves more stable pose estimation than handcrafted or regression-based alternatives. The fusion of LiDAR and vision improved robustness in scenes with low texture or poor illumination, validating the hypothesis that multi-modality can bridge the gap between accuracy and generalization. Importantly, the system produces interpretable outputs and operates within real-time constraints for retrieval, although further optimization is needed for pose estimation.

This thesis demonstrates that it is feasible to move beyond retrieval-only frameworks and provide full, explainable 6D poses suitable for SLAM. Future work should focus on improving runtime efficiency in the pose estimation module, incorporating more diverse datasets, and testing deployment on real robotic platforms. These developments could contribute to more autonomous and trustworthy robotic systems for exploration, disaster response, and agriculture in extreme environments.

Keywords

Multi-modal place recognition, Six degrees of freedom pose estimation, Simultaneous Localization and Mapping (SLAM) integration, Transformer-based encoders, Light Detection and Ranging (LiDAR), DINO version 2, SONATA, Feature aggregation, Sinkhorn Algorithm for Locally Aggregated Descriptors (SALAD), Unstructured planetary environments, Real-time retrieval

Sammanfattning

Autonom navigering i planetliknande miljöer innebär unika utmaningar på grund av avsaknad av GPS-signaler, begränsad semantisk struktur och visuell tvetydighet orsakad av repetitiva texturer eller svåra ljusförhållanden. Traditionella metoder för platsigenkänning och lokalisering förlitar sig antingen på täta kartor och strukturerade miljöer eller erbjuder endast grov återhämtning utan att uppskatta fullständiga 6-Degrees of Freedom (DoF) (sex frihetsgrader) poser. Detta begränsar deras användbarhet i realtids-SLAM (Simultaneous Localization and Mapping) för fältrobotik och planetutforskning.

Denna avhandling angriper problemet genom att utveckla ett multimodalt system som utför både platsigenkänning och relativ posuppskattning i ostrukturerade miljöer utan GNSS. Den föreslagna metoden kombinerar visuella egenskaper extraherade från en transformerbaserad kodare (DINOv2) med 3D-geometriska beskrivare från en LiDAR-baserad ryggrad (SONATA). Dessa egenskaper projiceras och justeras i 3D-rymden för att generera tolkbara korrespondenser, från vilka systemet uppskattar fullständiga 6D-poser. På återhämtningssidan aggregeras DINOv2-beskrivare med hjälp av SALAD, en inlärd VLAD-liknande modul, och söks effektivt med FAISS-indexering. Systemet utvärderas på Etna-vulkanens datamängd, som är representativ för planetära terränger.

Resultaten visar att den föreslagna modellen överträffar etablerade metoder för återhämtning såsom NetVLAD och TransVPR, samt uppnår mer stabil posuppskattning än handgjorda eller regressionsbaserade alternativ. Kombinationen av LiDAR och visuella data förbättrade robustheten i scener med låg textur eller dålig belysning, vilket bekräftar hypotesen att multimodalitet kan överbrygga gapet mellan noggrannhet och generalisering. Viktigt är att systemet genererar tolkbara resultat och fungerar inom realtidskrav för återhämtning, även om vidare optimering krävs för posuppskattningen.

Denna avhandling visar att det är möjligt att gå bortom enbart återhämtningsbaserade ramverk och tillhandahålla fullständiga, förklarliga 6D-poser som lämpar sig för SLAM. Framtida arbete bör fokusera på att förbättra prestandan i posuppskattningsmodulen, inkludera mer varierade datamängder och testa implementering på verkliga robotplattformar. Dessa framsteg kan bidra till mer autonoma och tillförlitliga robotsystem för utforskning, katastrofinsatser och jordbruk i extrema miljöer.

Nyckelord

Multi-modal platsigenkänning, Sex frihetsgraders posuppskattning, Integration av simultan lokalisering och kartläggning (SLAM), Transformerbaserade kodare, Ljusdetektering och avståndsmätning (LiDAR), DINO version 2, SONATA, Funktionell aggregering, Sinkhorn-algoritm för lokalt aggregerade beskrivare (SALAD), Ostrukturerade planetära miljöer, Återhämtning i realtid

Tiivistelmä

Autonominen navigointi planeettamaisissa ympäristöissä tuo mukanaan erityisiä haasteita GPS-signaalien puuttumisen, rajallisen semanttisen rakenteen sekä visuaalisen epäselvyyden vuoksi, jota aiheuttavat toistuvat tekstuurit ja vaikeat valaistusolosuhteet. Perinteiset paikan tunnistus- ja paikannusmenetelmät perustuvat joko tiheisiin karttoihin ja jäsenneltyihin ympäristöihin tai tarjoavat vain karkean haun ilman täysimääräistä 6-DoF (kuuden vapausasteen) asennon estimointia. Tämä rajoittaa niiden soveltuvuutta reaaliaikaiseen SLAM-järjestelmään (Simultaneous Localization and Mapping) kenttärobotiikassa ja planeettojen tutkimuksessa.

Tämä diplomityö käsittelee ongelmaa kehittämällä multimodaalisen järjestelmän, joka suorittaa sekä paikan tunnistusta että suhteellisen asennon estimointia jäsentymättömissä, GNSS-vapaissa ympäristöissä. Ehdotettu lähestymistapa yhdistää transformer-pohjaisesta kooderista (DINOv2) poimitut visuaaliset piirteet LiDAR-pohjaiseen runkoon (SONATA) perustuvien 3D-geometristen piirteiden kanssa. Nämä piirteet projisoidaan ja kohdistetaan 3D-avaruudessa tuottaen tulkittavia vastaavuuksia, joiden perusteella järjestelmä arvioi täydet 6D-asennot. Haun osalta DINOv2-piirteet yhdistetään SALAD-menetelmällä, joka on oppiva VLAD-tyylinen moduuli, ja haku toteutetaan tehokkaasti FAISS-indeksoinnin avulla. Järjestelmä arvioitiin Etna-tulivuoren tietoaineistolla, joka edustaa planeettamaista maastoa.

Tulokset osoittavat, että ehdotettu malli päihittää vakiintuneet hakumenetelmät kuten NetVLAD ja TransVPR, ja saavuttaa vakaamman asennon estimoinnin kuin käsintehdyt tai regressiopohjaiset vaihtoehdot. LiDARin ja visuaalisen tiedon yhdistäminen paransi järjestelmän kestävyyttä alhaisen tekstuurin tai heikon valaistuksen tilanteissa, vahvistaen hypoteesin siitä, että multimodaalisuus voi kuroa umpeen tarkkuuden ja yleistettävyyden välistä kuilua. Tärkeää on, että järjestelmä tuottaa tulkittavia tuloksia ja toimii reaaliaikaisissa hakuvaatimuksissa, vaikka asennon estimointimoduuli vaatii edelleen optimointia.

Tämä diplomityö osoittaa, että on mahdollista siirtyä pelkästään hakuun perustuvista järjestelmistä kohti täysiä, selitettävissä olevia 6D-asentoja, jotka soveltuvat SLAMiin. Tulevassa työssä tulisi keskittyä asennon estimoinnin suoritustehokkuuden parantamiseen, monipuolisempien tietoaineistojen käyttöönottoon sekä järjestelmän testaamiseen oikeilla robottialustoilla. Nämä kehitysaskeleet voivat edistää autonomisempien ja luotettavampien robottijärjestelmien kehitystä tutkimukseen, katastrofivalmiuteen ja maatalouteen äärimmäisissä olosuhteissa.

Avainsanat

Monimodaalinen paikantunnistus, Kuuden vapausasteen asentopositiomittaus, Samanaikainen paikannus ja kartoitus (SLAM) -integraatio, Transformer-pohjaiset kooderit, Valotutka (LiDAR), DINO versio 2, SONATA, Piirrekoosteet, Sinkhorn-algoritmi paikallisesti koottuja piirteitä varten (SALAD), Jäsentymättömät planetaariset ympäristöt, Reaaliaikainen haku

Acknowledgments

I would like to thank the German Aerospace Center (DLR) for hosting me during my thesis project and for providing financial support for my work. This work was supported by the Helmholtz Association project iFOODis (contract number KA2-HSC-06). It was an excellent environment to develop this research, both professionally and personally.

I would like to thank Riccardo Giubilato for his invaluable supervision at DLR. His continuous guidance, technical advice, and strategic suggestions were key throughout this project. His recommendations on what to implement, the regular discussions, and the data he provided for training and evaluating the models were essential for the development of the thesis.

I would like to thank John Folkesson for his supervision at KTH. His guidance and reading recommendations were helpful in shaping the direction of the work, and his support is appreciated.

I would also like to thank my colleagues and other interns at DLR, especially my office mates Kareem and Tommaso for the fun times, support, and memorable ping pong games that added balance to the hard work.

Finally, I want to express my deepest gratitude to my parents, Jose Antonio and Eugenia, and my boyfriend, Guglielmo, whose emotional and professional support was always present. Their belief in me and constant encouragement made this journey not only possible but enjoyable.

Munich, Germany, August 2025
Laura Alejandra Encinar Gonzalez

Contents

List of Figures

List of Tables

List of acronyms and abbreviations

ANN	Approximate Nearest Neighbor
BeV	Bird's-eye View
BoW	Bag-of-Words
BRIEF	Binary Robust Independent Elementary Features
CLS	Classification Token
CNN	Convolutional Neural Network
DLR	German Aerospace Center
DoF	Degrees of Freedom
DoG	Difference-of-Gaussian
DSAC	Differentiable RANSAC
ECA	Efficient Channel Attention
EMM	Essential Matrix Module
FAISS	Facebook AI Similarity Search
FOV	Field of View
FPFH	Fast Point Feature Histograms
FPN	Feature Pyramid Network
GAP	Global Average Pooling
GeM	Generalized Mean Pooling
GGeM	Group Generalized Mean Pooling
GMP	Global Max Pooling
GNN	Graph Neural Network
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
HLoc	Hierarchical Localization
HNSW	Hierarchical Navigable Small World graphs
ICP	Iterative Closest Point
IMU	Inertial Measurement Unit
IVF	inverted file

LiDAR	Light Detection and Ranging
LoFTR	Local Feature TRansformer
LRU	Lightweight Rover Unit
LSH	Locality-Sensitive Hashing
MAE	Masked Autoencoder
MLP	Multi-layer Perceptrons
NN	Nearest Neighbors
ORB	Oriented FAST and Rotated BRIEF
PCA	Principal Component Analysis
PnP	Perspective-n-Point
PQ	product quantization
RANSAC	RANdom SAmple Consensus
RPR	relative pose regression
SDG	Sustainable Development Goals
SfM	Structure-from-Motion
SIFT	Scale-Invariant Feature Transform
SLAM	Simultaneous Localization and Mapping
SURF	Speeded-Up Robust Features
UMF	Unifying Local and Global Multi-modal Features
UN	United Nations
VIO	Visual-Inertial Odometry
ViT	Vision Transformer
VLAD	Vector of Locally Aggregated Descriptors
VPR	Visual Place Recognition

Chapter 1

Introduction

This chapter introduces the foundations and motivations behind the thesis. It begins with a background on visual place recognition, pose estimation, and their role in Simultaneous Localization and Mapping (SLAM), particularly in Global Navigation Satellite System (GNSS)-denied and unstructured environments. Section 1.2 presents the core problem and research question, outlining the limitations of current methods in planetary-like terrains. Section 1.2.1 defines the original problem in detail and identifies the key scientific and engineering challenges. Section 1.3 clarifies the thesis purpose within a broader societal and research context, followed by a breakdown of the project goals in Section 1.4. The methodology adopted to address these challenges is described in Section 1.5. Section 1.7 discusses the thesis' ethical, environmental, and social considerations. Finally, Section 1.8 outlines the structure of the rest of the thesis.

1.1 Background

Place recognition and pose estimation are critical components of robotic navigation systems, particularly in **SLAM**, where they enable loop closure detection to improve localization accuracy. The seminal tutorial on SLAM highlights the importance of loop closure detection in achieving consistent maps and precise localization [1]. In planetary exploration scenarios, where GNSS are unavailable, robust place recognition and pose estimation ensure accurate navigation in unstructured environments.

Traditional visual **place recognition** methods rely heavily on image-based techniques that identify similar places by comparing visual features. Classical approaches such as ORB-SLAM3 [2], an open-source SLAM

system, utilize binary descriptors and kd-tree-based indexing to detect loop closures efficiently [3, 4]. While effective in structured environments, they struggle in planetary-like terrains, where the absence of distinct features or extreme changes in appearance can degrade performance. For example, keypoint-based methods like Scale-Invariant Feature Transform (SIFT) [5], ORB [6], and SuperPoint [7] depend on distinct visual features, making them less suitable for featureless landscapes. In contrast, detector-free methods like Local Feature TRansformer (LoFTR) [8] do not begin by detecting discrete keypoints, but instead extract features for every pixel or patch. This approach enables the model to identify matches even in low-texture or repetitive areas where keypoint-based detectors typically fail. However, their application to extreme environments remains underexplored.

Deep learning approaches like NetVLAD and Hierarchical Localization have advanced visual place recognition by learning robust global descriptors and generating 6D pose outputs [9, 10]. Despite their advancements, these methods are computationally intensive, making them unsuitable for real-time applications on resource-constrained systems like planetary rovers. To address these challenges, **multi-modal approaches** that integrate vision and LiDAR data have gained traction. Vision provides rich texture and color information, while LiDAR offers robust geometric insights, even in low-light conditions. Recent research, such as the method proposed in [11], tackles the challenge of place recognition in low-texture environments as an image retrieval problem, achieving better accuracy than unimodal methods like PointNet or NetVLAD and outperforming previous multimodal approaches like AdaFusion [12, 9, 13]. However, it does not tackle the problem of **pose estimation**, which is crucial for integrating place recognition into SLAM.

Pose estimation in **feature-deficient environments** is typically approached via end-to-end learning-based methods or structure-based techniques. End-to-end learning-based approaches like PoseNet estimate camera pose directly from images but may lack the precision of geometric techniques [14]. Structure-based techniques such as Perspective-n-Point (PnP) with RANdom SAmple Consensus (RANSAC) estimate camera pose using 2D-3D correspondences but require sufficient keypoints for reliability [15, 16]. Hybrid approaches, including Structure-from-Motion (SfM) pipelines like COLMAP and Iterative Closest Point (ICP) [17, 18], leverage 3D structure for pose refinement and can integrate LiDAR data for improved accuracy in textureless environments. However, traditional SfM approaches are highly data-intensive and computationally expensive, requiring multiple images from different viewpoints to reconstruct the 3D structure. This makes them

impractical for real-time rover applications in planetary-like terrains, where obtaining diverse viewpoints may be infeasible.

The primary objective of this thesis is to bridge the gap between place recognition and SLAM by integrating pose estimation into a multi-modal framework that leverages both image and LiDAR data. The proposed algorithm will focus on **recognizing places and estimating the relative pose of the camera**, thereby enhancing the localization and mapping capabilities of the Lightweight Rover Unit (LRU) **in unstructured, planetary-like environments** [19]. By combining the complementary strengths of vision and LiDAR, this research aims to develop a robust and computationally efficient solution for real-time navigation in extreme and feature-deficient terrains.

1.2 Problem

In visual SLAM systems, closing a loop does not only require identifying a previously visited place, it also requires estimating the **relative pose** between the current observation and the matched location. This transformation is essential to correct accumulated drift and ensure map consistency. Despite its importance, many existing methods treat place recognition purely as an **image retrieval problem** [11], ignoring the estimation of relative spatial transformation between the retrieved image and the query.

Additionally, most prior work in place recognition and pose estimation is developed for **structured environments**, where texture-rich scenes and clearly defined landmarks simplify keypoint extraction and matching. In contrast, **planetary-like terrains** characterized by low texture, repetitive features, and sparse geometry—pose a significantly greater challenge. Traditional feature-based methods often fail in these settings due to the absence of distinctive visual cues.

This thesis addresses both challenges: first, by integrating **relative pose estimation** into the place recognition pipeline, and second, by developing methods that remain effective in **unstructured, feature-sparse environments**. To improve robustness, the project also explores **multi-modal integration**, combining vision and LiDAR to compensate for the limitations of each modality.

How can autonomous systems reliably estimate relative pose and perform robust loop closure in low-texture, planetary-like environments where conventional image-based methods fail?

1.2.1 Original problem and definition

The original problem addressed in this thesis stems from the limitations of current visual place recognition systems in **unstructured, planetary-like environments**. Existing methods are primarily designed for structured scenes and focus on recognizing previously visited places, often without providing an associated **relative pose estimate**. As a result, they cannot be directly integrated into SLAM frameworks, which require both recognition and geometric alignment to perform loop closure and correct drift.

The core challenge is to develop a **multi-modal place recognition** system that not only retrieves relevant past observations but also computes a reliable 6-Degrees of Freedom (DoF) pose transformation between them. This must be achieved under the constraints of real-time operation and in terrains where classical visual features are sparse or non-discriminative. The solution must be scalable, data-efficient, and suitable for deployment on computationally constrained platforms such as the **LRU**, a robot available at German Aerospace Center (DLR) designed for planetary exploration [19].

1.2.2 Scientific and engineering issues

Several scientific and engineering challenges must be addressed to solve this problem effectively. From a scientific perspective, the lack of texture and structure in planetary-like environments raises open questions about how to extract **discriminative and robust visual descriptors** that generalize beyond the training domain. Estimating relative pose from visual data remains difficult in the absence of stable keypoints or dense geometric information, particularly in scenes dominated by repetitive or ambiguous features like rock fields and horizon lines.

A further scientific challenge lies in defining what constitutes a **valid loop closure** during training and evaluation. Establishing **ground truth similarity** between image pairs typically relies on the camera's position and orientation, assuming a well-calibrated and synchronized dataset. This is especially critical when integrating LiDAR, which requires tight **sensor synchronization and alignment**. Even under these conditions, inconsistencies arise: images taken from the same location may differ significantly due to occlusions or dynamic changes in the environment, making them hard for a deep model to associate. Conversely, images captured from distant viewpoints but facing similar terrain (e.g., the same horizon line) may appear deceptively similar to a deep network despite large spatial separation. These ambiguities complicate the creation of reliable training

signals and performance benchmarks.

From an engineering standpoint, the solution must operate in **real-time** on platforms with **limited computational resources**, such as the LRU [19]. This necessitates the use of computationally efficient architectures and careful trade-offs between model complexity and responsiveness.

1.3 Purpose

The purpose of this thesis is to develop a **robust and efficient system for multi-modal place recognition and relative pose estimation in unstructured, planetary-like environments**, with a focus on improving loop closure detection in SLAM pipelines. The project aims to bridge the current gap between image-based place recognition and the geometric requirements of SLAM by integrating pose estimation into a learning-based, multi-modal framework that combines visual and LiDAR data.

The broader purpose of the degree project is to contribute practical and scientifically validated solutions to the domain of autonomous navigation in extreme and **feature-sparse environments**, such as those encountered in planetary exploration. By developing methods that can generalize to low-texture, ambiguous terrains where traditional techniques fail, this work supports ongoing efforts in robotics and space exploration. Specifically, the research will benefit institutions like the **German Aerospace Center (DLR)** [20], which deploys systems such as LRU for testing autonomous capabilities in Mars-analogue scenarios [19].

The societal and scientific value of the work lies in enabling more robust, resource-efficient, and autonomous robotic systems. In space robotics, greater autonomy reduces the need for human intervention, which is essential for long-duration missions and improves mission safety. On Earth, similar systems can be applied to disaster response, agriculture, and search-and-rescue in areas where GPS is unreliable and human access is limited.

From an ethical and sustainability perspective, the project supports **responsible AI deployment** in critical systems by focusing on transparency, robustness, and reliability. The use of open datasets and reproducible methods contributes to scientific integrity. Moreover, by enabling more effective and autonomous robotic systems, the project indirectly contributes to sustainable space exploration, minimizing reliance on energy-intensive human supervision and improving the operational lifespan of robotic explorers.

1.4 Goals

The primary goal of this thesis is to **develop a multi-modal place recognition and pose estimation system** capable of operating in unstructured, planetary-like environments. The aim is to go beyond image retrieval by producing 6D relative pose estimates suitable for integration into SLAM pipelines, while ensuring the approach is efficient enough for real-time deployment on systems like the LRU.

To fulfill this goal, the work has been structured into the following three sub-goals:

1. **Evaluate the benefits of multi-modality (vision + LiDAR) in place recognition and pose estimation** This involves reviewing state-of-the-art multi-modal learning techniques, selecting suitable datasets representative of unstructured terrains, and analyzing the impact of combining visual and LiDAR inputs. Tasks include preprocessing synchronized data, implementing baseline models, and comparing their performance to quantify the contribution of LiDAR in environments where visual features alone are insufficient.

2. **Develop an algorithm that outputs 6D poses for SLAM integration instead of simple image retrieval** The focus is on extracting and combining descriptors from both modalities in a geometrically consistent way and producing full 6-DoF transformations between query and matched observations. This goal includes benchmarking against existing approaches, defining robust evaluation metrics, and validating the model in real planetary-like conditions.

3. **Optimize computational and memory efficiency by integrating model-based components into deep learning-based multi-modal methods** This includes identifying classical geometric techniques or SLAM priors that can reduce reliance on high-capacity deep models, and investigating inference-time optimization methods such as quantization, pruning, or knowledge distillation. Testing will be conducted under hardware constraints similar to those of the LRU to ensure real-world feasibility.

1.5 Research Methodology

This thesis adopts an applied experimental research approach grounded in engineering problem-solving. The project is driven by a practical need: enabling accurate and efficient place recognition and pose estimation in unstructured, planetary-like environments. To address this, the methodology combines deep learning-based algorithm development, comparative evaluation, and multi-modal data integration, supported by an experimental framework and benchmark analysis.

At the core of the project is the design of a **deep learning-based algorithm for multi-modal place recognition and pose estimation**, integrating both **visual (RGB images)** and **geometric (LiDAR point clouds)** data. The research follows a design science methodology, where the main goal is to iteratively develop, implement, and evaluate a working artifact that meets clearly defined performance criteria. The system will be evaluated not only for recognition accuracy but also for computational efficiency and pose estimation quality—metrics aligned with the constraints of autonomous planetary rovers.

The project explores **hybrid architectures** that incorporate both **learned and model-based components**. Deep feature extractors (transformers) are used to obtain discriminative descriptors from images and point clouds, while classical model-based techniques (such as PnP or ICP) support geometric consistency and pose estimation. This hybridization approach was chosen to balance data-driven generalization with the interpretability and structure of geometric models, particularly important in environments where texture is sparse and viewpoints vary significantly.

Several methodological alternatives were considered. Purely geometric methods were excluded due to their reliance on stable keypoints and dense 3D structure, which are not available in the target environments. End-to-end pose regression networks were also considered but deprioritized, as they often struggle with generalization across terrain types and fail to capture the spatial consistency required for SLAM integration. Instead, the chosen approach leverages the modularity and interpretability of **descriptor-based retrieval** and **relative pose estimation** from correspondences.

In terms of philosophical assumptions, the thesis is grounded in a realist paradigm: it assumes that camera pose and physical structure exist independently of observation and can be measured or estimated through sensing. The approach is deductive, testing the performance of the designed system against established SLAM objectives and metrics. The methodology emphasizes **empirical validation** through experiments on publicly available

datasets and real-world rover trials.

Three primary methodological phases guide the project:

- **Algorithm Development**: Design and implement a multi-modal deep learning pipeline using transformer-based feature extractors (e.g., DINOv2 for images, Sonata for point clouds). Descriptors from each modality are aggregated and fused to allow fast and robust place recognition and relative pose estimation. Aggregation methods like SALAD are used to compress image features into compact, comparable descriptors.

- **Dataset Preparation**: Utilize publicly available datasets such as the *Etna dataset* and DLR's outdoor rover trials [21]. If needed, additional data from the Morocco-Acquired dataset of Mars-Analog eXploration (MADMAX) may be included to increase environmental diversity [22].

- **Evaluation**: Conduct systematic experiments to assess the system's performance. Metrics include precision and recall for place recognition, yaw error for pose estimation, and inference speed for real-time deployment feasibility.

This methodology supports both the scientific exploration of novel feature fusion strategies and the engineering validation of a system intended for deployment on real-world robotic platforms.

1.6 Delimitations

This thesis focuses on the development of a multi-modal place recognition and 6D pose estimation system using vision and LiDAR data in unstructured, planetary-like environments. However, there are several important delimitations that define the scope and boundaries of this work.

First, the project **does not involve the development of a complete SLAM** system. While the proposed method is designed to serve as a component within the **loop closure detection** module of a SLAM pipeline, the thesis does not implement or evaluate integration with full SLAM frameworks. All SLAM-related processes—such as map building, odometry correction, or back-end optimization—are considered out of scope.

Second, **real-time deployment on robotic platforms**, such as the LRU, is not included in this project due to time and resource constraints. Although the system is designed with computational efficiency in mind and is intended

for future integration into the LRU, this thesis focuses exclusively on **offline testing using recorded datasets**. Evaluation will be limited to metrics such as recognition precision, recall, and yaw error for pose estimation, without real-time testing or performance validation on embedded hardware.

These delimitations ensure that the thesis remains focused on the core research contributions: improving robustness and accuracy in place recognition and pose estimation through multi-modal deep learning, without extending into full SLAM system development or deployment.

1.7 Ethical, Social, and Sustainability Considerations

In addition to addressing technical challenges, this thesis places strong emphasis on ethical responsibility, sustainability, and social relevance, especially as autonomous systems are increasingly deployed in critical real-world applications.

The proposed methodology is fully reproducible, transparent, and ethically sound. All datasets used during training and evaluation are publicly available and licensed for academic use and redistribution. The primary evaluation was conducted on the Etna dataset [21], a benchmark collected in a planetary-analog environment on Mount Etna, Sicily. This dataset includes grayscale images, LiDAR point clouds, and D-GNSS ground truth poses. It contains no human subjects or sensitive information, thus posing no privacy risks or ethical concerns. Additional training data for the image retrieval module was sourced from urban-scale benchmarks—Mapillary Street-Level Sequences (MSLS) [23], GSV Cities [24], and Pittsburgh250k [25]—all of which contain public street imagery captured in outdoor settings, with no manual annotations or personally identifiable content.

From a technical reproducibility perspective, all aspects of the system—data preprocessing, model training, evaluation, and benchmarking—are implemented using open-source Python scripts. Full reproducibility was a guiding principle throughout the project: every experiment is scriptable, parameterized, and designed to be rerun on any system with appropriate hardware. Upon submission, the entire codebase and trained model checkpoints will be released publicly, supporting the broader research community in building on this work.

Beyond these research norms, the thesis aligns with broader sustainability goals. Efficient and modular visual-LiDAR localization systems have the

potential to reduce the energy footprint of autonomous agents by enabling real-time, onboard processing without reliance on cloud infrastructure. This is particularly valuable in planetary exploration, where communication latency and bandwidth are extremely limited. The use of compact descriptors like SALAD [26], and efficient indexing (e.g., FAISS [27]) reflects a conscious effort to design methods that are not only accurate but computationally efficient, making them more deployable in low-power robotic systems.

This focus on computational efficiency is especially relevant in planetary exploration, where communication delays, bandwidth limitations, and power constraints demand fully autonomous systems that operate reliably and independently. Beyond its technical challenges, planetary exploration carries significant societal and scientific importance: it drives technological innovation, fosters international cooperation, and expands our understanding of Earth's place in the cosmos. Enabling robust, interpretable localization in these missions directly supports this broader vision—ensuring that robotic platforms can safely navigate, collect data, and carry out scientific objectives in extreme and unstructured environments. By advancing autonomy in these contexts, the system contributes not only to mission success but also to the long-term goal of sustainable and responsible exploration beyond Earth.

Beyond space missions, the system contributes to other socially beneficial domains such as disaster response, environmental monitoring, and precision agriculture. These applications often share similar constraints—unstructured environments, poor lighting, and lack of infrastructure—making the developed techniques broadly applicable. Improving localization robustness and explainability in such contexts can accelerate rescue missions, enhance environmental resilience, and expand the frontier of autonomous robotics in both Earth and space.

Finally, the system is explicitly designed for interpretability and transparency, in contrast to many black-box AI methods. By using matching-based pose estimation and modular components, the pipeline enables users to understand, validate, and diagnose localization decisions, an essential property for ethical deployment in high-stakes scenarios. This commitment to explainability aligns with increasing demands for accountability in AI and robotics, especially as these technologies enter socially and environmentally sensitive domains.

1.8 Structure of the thesis

This thesis is organized into seven chapters. Chapter 1 introduces the background, problem formulation, purpose, goals, research methodology, and delimitations of the project. Chapter 2 presents the necessary background on SLAM, visual place recognition, pose estimation, and multi-modal approaches, followed by a review of related work in the field. Chapter 3 will describe the research methods and experimental design used to develop and evaluate the proposed approach. Chapters 4 to 7 will cover the system implementation, experimental results, discussion, and final conclusions, including reflections and directions for future work.

Chapter 2

Background

This chapter provides essential background information on visual place recognition and pose estimation, with a focus on their roles in Simultaneous Localization and Mapping (SLAM) systems. It introduces key challenges specific to unstructured, planetary-like environments, where traditional methods often fail due to low texture and sparse visual cues. In addition, this chapter outlines the components and phases of a place recognition pipeline, including feature extraction, descriptor aggregation, and pose estimation. Finally, the chapter presents a review of related work in multi-modal learning, transformer-based representation learning, and LiDAR-based registration, highlighting recent advancements and identifying gaps that motivate the present study.

2.1 Fundamentals of SLAM and Loop Closure Detection

2.1.1 SLAM and its Significance in Autonomous Navigation

SLAM refers to the process by which a mobile robot concurrently constructs a model of its environment while estimating its own position within that environment. The primary objective of SLAM is to enable a robot to navigate autonomously in an unknown environment without relying on Global Positioning System (GPS). Instead, the robot builds a map incrementally using sensor data and updates its localization within that map over time [1].

SLAM is a fundamental component of autonomous navigation, allowing

robots to explore unstructured environments, from indoor spaces to large-scale outdoor settings, including planetary surfaces. Without SLAM, navigation systems must rely solely on odometry, which accumulates drift over time. The incorporation of mapping allows for the correction of localization errors through loop closure detection, significantly improving the accuracy and robustness of navigation [1].

2.1.2 The Role of Place Recognition and Pose Estimation

Place recognition and pose estimation play a critical role in ensuring the consistency of the SLAM-generated map [28]. Place recognition allows the robot to recognize previously visited locations, even if they are viewed from different angles or under different conditions. Once a place is recognized, relative pose estimation determines how the robot's current viewpoint relates to the previously recorded scene.

These components are essential for loop closure detection, where the system identifies that the robot has returned to a previously mapped area and corrects localization drift. However, in feature-sparse environments such as deserts, planetary surfaces, or underwater regions, place recognition and pose estimation remain difficult due to the lack of distinct visual landmarks [1].

2.1.3 Challenges of Loop Closure in Planetary Environments

Loop closure detection becomes particularly challenging in planetary exploration scenarios due to the following constraints:

- **GNSS Absence:** Unlike terrestrial robots, planetary rovers cannot rely on satellite-based positioning, requiring SLAM to rely entirely on visual, Light Detection and Ranging (LiDAR), or inertial sensors for localization.
- **Low-Texture Environments:** Deserts, volcanic landscapes, and extraterrestrial terrains are characterized by large regions of uniform sand or rock, offering few discriminative features for place recognition [1].
- **Limited Landmarks:** Natural landmarks such as craters and mountains are typically distant from the rover's camera, making accurate pose estimation challenging.

Addressing these challenges requires novel approaches, including self-supervised deep learning for feature extraction, multimodal data fusion (vision and LiDAR), and advanced loop closure verification techniques to minimize false matches.

2.2 Visual Place Recognition

Visual Place Recognition (VPR) is a critical component of SLAM, enabling robots to recognize previously visited locations and correct accumulated drift through loop closure detection. This capability is particularly important in GPS-denied or perceptually ambiguous environments, such as planetary surfaces, caves, or underwater terrains. To address these challenges effectively, VPR pipelines are typically divided into distinct phases, with specialized methods tailored to each stage. The following sections review the key techniques employed at each phase of the VPR process.

2.2.1 Feature Extraction

The first and most critical phase of VPR is feature extraction—the process of extracting meaningful representations from raw images using either handcrafted or learned descriptors. These features must be robust to changes in viewpoint, lighting, and environmental texture, particularly in planetary or extreme environments where traditional navigation cues are minimal. VPR methods can be categorized according to their feature extraction approach, broadly divided into classical techniques and learning-based methods, each with distinct advantages and limitations.

a) Classical Methods

Classical VPR methods rely on handcrafted features and descriptors to match images across different viewpoints and conditions. These approaches, while computationally efficient, often struggle in feature-deficient environments such as planetary landscapes.

One of the most well-known methods, **SIFT** [5], extracts keypoints that are invariant to scale and rotation using Difference-of-Gaussian (DoG) filters. Although robust to viewpoint changes, SIFT relies heavily on high-gradient features, making it less effective in low-texture terrains such as sandy or volcanic landscapes. **Speeded-Up Robust Features (SURF)** improves detection speed by using Haar wavelet

approximations and integral images [29]. While faster than SIFT, its dependence on local texture patterns similarly limits its effectiveness in low-contrast planetary terrains where distinctive features are rare.

Oriented FAST and Rotated BRIEF (ORB) enhances computational efficiency by combining FAST keypoints with Binary Robust Independent Elementary Features (BRIEF) descriptors [6]. This method is particularly useful in real-time applications, as demonstrated in ORB-SLAM3 [2], which utilizes ORB features for simultaneous localization and mapping (SLAM). However, its performance degrades in environments with repetitive or sparse features.

A major limitation of these classical methods in planetary environments is perceptual aliasing, where different locations appear visually similar, leading to incorrect matches. The survey by Barros et al. [30], highlights that handcrafted descriptors such as SIFT and SURF are particularly sensitive to these conditions [5, 6].

b) Learning-based Methods

Deep learning has transformed VPR by enabling systems to learn discriminative and invariant feature representations directly from raw image data. Unlike handcrafted approaches, learning-based methods adapt better to changes in lighting, viewpoint, and environmental structure. Their development has followed a progression from Convolutional Neural Network (CNN) based models to transformer-based self-supervised architectures.

NetVLAD was one of the earliest learning-based approaches in VPR [9]. It extends the traditional Vector of Locally Aggregated Descriptors (VLAD) framework by learning a differentiable pooling layer that aggregates local CNN features into a global vector. While highly effective in structured urban environments, it requires extensive labeled data and struggles with extreme viewpoint changes. **DenseVLAD** improves robustness in low-texture environments by incorporating dense feature extraction [31], though it remains vulnerable to domain shifts, such as applying Earth-trained models to Martian landscapes. **Patch-NetVLAD** enhances recognition of distant landmarks by incorporating multi-scale feature aggregation [32], making it more applicable to planetary horizons. However, all of these CNN-based approaches depend on large labeled datasets, which limits their applicability in environments where annotated data is unavailable.

Methods like **LoFTR** [8] extend the capabilities of CNN-based extractors by introducing transformer-based modules to enhance feature representations. LoFTR first extracts coarse and fine features using a convolutional backbone with a Feature Pyramid Network (FPN), leveraging CNNs' local inductive bias and computational efficiency. These features are then passed through a transformer module that encodes position- and context-aware representations using attention mechanisms. Although primarily used for dense matching tasks, LoFTR's approach illustrates how transformer-based refinement of CNN features can improve robustness in unstructured or low-texture environments.

Recent developments have led to fully transformer-based models designed for general-purpose feature extraction. Models such as **DINO** and its successor **DINOv2** utilize self-supervised learning and Vision Transformers (ViTs) to produce general-purpose features without requiring annotated datasets [33, 34]. These models are trained using a teacher-student distillation framework that captures both semantic and structural patterns at multiple spatial resolutions. DINOv2, in particular, has demonstrated strong transferability across tasks and environments without the need for fine-tuning, making it suitable for deployments where labeled data collection is impractical. Research shows that features from intermediate transformer layers often contain richer positional information than those from the final layer [34], which can enhance performance in place recognition tasks, especially in unstructured or perceptually degraded environments.

In summary, the shift from CNN-based to transformer-based and self-supervised feature extractors reflects a growing need for generalization, scalability, and data efficiency in VPR. These methods improve robustness in challenging environments, though issues such as computational cost and domain transferability remain key challenges for future research.

2.2.2 Feature Aggregation

After extracting features from images, VPR systems aggregate these features into compact and discriminative descriptors for efficient comparison. The choice of aggregation strategy significantly impacts the descriptor's quality, size, and the system's overall performance.

a) Attention-Based Aggregation

Transformer-based models like DINO and DINOv2 rely on a Classification Token (CLS) to aggregate information from all image patches [33, 34]. The **CLS token** is a learned vector that gathers global context through multiple attention layers. It is commonly used as the image-level descriptor in many vision transformer models. However, studies such as AnyLoc have shown that relying solely on the CLS token may be suboptimal for place recognition in unstructured environments [35]. Instead, aggregating local features directly can yield better performance in such settings.

Beyond the CLS token, other models implement more sophisticated attention-based aggregation mechanisms. **TransVPR** [36], for example, extracts multi-level features from different layers of a transformer and combines them using attention across spatial scales. This approach captures both low-level details and high-level semantic information, resulting in a more expressive global descriptor. In addition to producing a global vector, TransVPR also generates patch-level descriptors that enable geometric verification, an important step for reducing false positives in place recognition.

Attention-based aggregation dynamically focuses on informative image regions, improving robustness in complex scenes, but often comes with higher computational cost compared to simpler methods.

b) MLP Mixer-Based Aggregation

Multi-layer Perceptrons (MLP) Mixers offer an alternative to attention mechanisms for combining spatial and channel-wise information across feature maps [37]. These models use stacked MLP to mix features across tokens and channels, capturing spatial relationships without relying on self-attention.

In **DinoMix** [38], the final transformer layer output from DINOv2 is passed through an MLP Mixer composed of consecutive MLP blocks and linear projections. The goal is to extract high-level spatial relationships and produce a compact global descriptor. A similar design is used in **MixVPR** [39], which applies the MLP Mixer on CNN-derived features instead of transformer outputs. MixVPR has shown competitive performance, outperforming methods such as Patch-NetVLAD, TransVPR, and SuperGlue in certain benchmarks[32, 36, 40].

The optimal architecture for the mixer, based on empirical evaluations [41], consists of two mixing layers. This configuration achieves a good trade-off between expressiveness and computational efficiency. However, its effectiveness may vary across datasets, and in some cases, it performs slightly below more complex aggregation schemes like VLAD.

c) Pooling-Based Aggregation

Pooling methods represent a simpler and more efficient class of aggregation strategies. These include **Global Average Pooling (GAP)** [42], **Global Max Pooling (GMP)** [43], and **Generalized Mean Pooling (GeM)** [44]. GAP computes the average value of each feature map, GMP selects the maximum, and GeM introduces a learnable parameter that generalizes both operations. Variants such as **Group Generalized Mean Pooling (GGeM)** further improve upon this by dividing feature channels into groups and applying different pooling parameters to each [45], emphasizing important features while suppressing trivial ones. The AnyLoc study provides a comprehensive comparison of these pooling strategies when applied to DINO features [35], finding that while GeM offers a good balance between performance, speed and memory efficiency, it is often outperformed by more complex aggregation schemes such as VLAD in unstructured or low-texture environments.

d) VLAD-Based Aggregation

VLAD-based aggregation remains one of the most powerful approaches for place recognition. The original VLAD algorithm aggregates features by assigning them to pre-defined cluster centers and computing the residuals between each feature and its assigned centroid [46]. These residuals are then concatenated into a global descriptor. **NetVLAD** extends this concept by learning the cluster centers and using soft assignment instead of hard clustering, allowing for end-to-end training [9]. **Patch-NetVLAD** builds further on this idea by incorporating multi-scale features and introducing a geometric consistency measure known as Rapid Spatial Scoring [32, 47], which compares vertical and horizontal distances between matched patch positions in two images. These refinements improve recognition of distant landmarks and increase robustness to spatial distortions.

A notable advancement in this area is **SALAD** [26], which inte-

grates DINO features with a modified VLAD framework. Unlike NetVLAD, which initializes its cluster centers from precomputed k-means centroids, SALAD learns its cluster assignments using two fully connected layers. It also introduces a "dustbin" cluster to discard uninformative features, reducing noise in the final descriptor. Feature-to-cluster assignment is reformulated as an optimal transport problem, and SALAD applies the Sinkhorn algorithm to obtain the optimal soft assignment [48]. Instead of computing residuals, features assigned to each cluster are summed and processed through fully connected layers for dimensionality reduction. Finally, SALAD concatenates this cluster-based representation with a global DINO token for improved semantic encoding. This combination has demonstrated strong performance, particularly under challenging conditions such as severe viewpoint or lighting changes, and outperforms other DINO-based aggregation methods with similar descriptor sizes such as GeM or MixVPR [44, 39].

In summary, feature aggregation defines how extracted features are combined into a representation suitable for place recognition. While attention-based methods offer dynamic, high-capacity representations, simpler techniques like pooling remain useful in low-resource contexts. Mixers offer a simple yet effective solution, while VLAD-based methods provide stronger feature discrimination, particularly when enhanced with self-supervised backbones like DINO. The selection of an aggregation method depends on the trade-off between robustness, efficiency, and environmental complexity.

2.2.3 Database indexing

In VPR, once image descriptors are extracted, they must be organized into efficient data structures to enable rapid and accurate retrieval. This process, known as database indexing, is crucial for scaling VPR systems to large environments. Below, we delve into various indexing methods.

a) Exact Nearest Neighbor and k-d Tree Search

The most straightforward retrieval method is exact **Nearest Neighbors (NN)** search [49], where a query descriptor is compared against every descriptor in the database to find the most similar one. While this **brute-force** method guarantees maximum accuracy, it becomes computationally impractical as the database size grows, especially when dealing with high-dimensional descriptors commonly used in VPR.

To improve search speed, data structures like **k-d trees** have been introduced [50]. K-d trees organize descriptors into a binary tree based on recursive partitioning of the feature space. This allows for efficient pruning during search, significantly reducing the number of comparisons required. However, their performance degrades rapidly in high-dimensional spaces, where most points become nearly equidistant, and the tree structure provides little benefit. As a result, k-d trees are suitable for small or low-dimensional datasets but are rarely used in recent large-scale VPR applications.

b) Approximate Nearest Neighbor (ANN) Search and FAISS

To overcome the limitations of exact methods, **Approximate Nearest Neighbor (ANN)** algorithms allow a controlled loss in retrieval accuracy in exchange for faster search times [51]. These methods are especially useful for high-dimensional data and large databases, where exact search becomes infeasible.

One widely adopted ANN solution is **Facebook AI Similarity Search (FAISS)**, an open-source library that supports multiple indexing techniques optimized for dense vector retrieval [27]. FAISS includes both CPU and GPU implementations and offers a variety of indexing structures such as flat (brute-force), inverted file (IVF) indexing, product quantization (PQ), and Hierarchical Navigable Small World graphs (HNSW). For example, IVF partitions the database into coarse clusters to narrow down the search space, while PQ compresses vectors into low-bit representations to reduce memory usage. HNSW, a graph-based approach, provides highly efficient search by constructing a multi-layer proximity graph.

The key advantage of FAISS is its flexibility: users can choose an index that balances speed, memory efficiency, and accuracy according to their needs. However, approximate methods inherently involve a trade-off in precision, and tuning FAISS parameters requires care to avoid significant performance loss. Despite this, FAISS has become a standard tool in large-scale VPR pipelines due to its scalability and robustness.

c) Bag-of-Words (BoW) Models

Another common indexing approach in visual recognition is the **Bag-of-Words (BoW)** model [52]. Inspired by text retrieval, BoW treats visual

features as ”words” by assigning local descriptors to entries in a pre-trained visual vocabulary (typically generated via **k-means clustering**) [3]. Each image is then represented as a histogram of word frequencies, which can be compared efficiently using inverted indices.

BoW offers a compact and computationally efficient representation, making it attractive for VPR systems and scenarios where real-time performance is essential. Its storage requirements are low, and retrieval can be performed quickly using established information retrieval techniques. However, BoW suffers from several limitations: it discards spatial information, making it less effective in scenes where layout matters, and its accuracy is highly dependent on the quality and size of the vocabulary.

d) Hashing-Based Indexing

Hashing methods project high-dimensional descriptors into a lower-dimensional binary space, allowing similarity comparisons using Hamming distance. Examples include **Locality-Sensitive Hashing (LSH)** and **Spectral Hashing**, which aim to ensure that similar descriptors hash to similar binary codes [53, 54].

The primary benefit of hashing is its high speed and low memory footprint. Binary representations allow for rapid bit-wise comparisons and can be stored compactly, which is advantageous for large-scale or embedded applications. However, hashing often leads to information loss, especially when the binary representation is too compact or the hash function is poorly aligned with the feature distribution. This can result in lower accuracy and missed matches, particularly in environments with subtle visual differences or repeated structures.

2.2.4 Place Matching

After feature extraction and indexing, the next step in the VPR pipeline is place matching, identifying which entries in the database most closely correspond to a query image. This stage can be divided into two levels: global image matching, where compact descriptors are compared to identify likely candidates, and local correspondence matching, where spatial consistency is verified through keypoint or dense feature alignment.

At the **global level**, similarity between image descriptors is typically measured using distance metrics such as Euclidean distance or **cosine similarity**.Euclidean distance measures the straight-line distance between two

points in feature space and is sensitive to descriptor magnitude. Cosine similarity, on the other hand, compares the angle between descriptors, normalizing for their length and focusing on their direction. It is particularly useful when descriptors are normalized to unit vectors. In practice, a match is usually considered valid if it exceeds a similarity threshold or ranks within the top-k nearest neighbors. A widely used method to reduce false positives is **Lowe's ratio** test [55], which compares the distance of the closest match to the second-closest. A match is accepted only if the ratio between these two distances is below a set threshold (typically 0.7), indicating that the best match is significantly better than the next best candidate and thus more likely to be correct.

However, image-level similarity alone may not be sufficient in visually ambiguous or feature-sparse environments. To improve both robustness and efficiency, many VPR systems adopt a hierachical frameworks such as **Hierarchical Localization (HLoc)** [10], which combines global retrieval with local geometric verification. This two-stage methods enable scalable localization in large environments by first narrowing down potential matches using global descriptors and then refining pose estimates with local features. HLoc has been widely explored for terrestrial applications and hold potential for planetary exploration, where extreme viewpoint variations and feature sparsity can make single-stage approaches unreliable. The survey by Barros et al. highlights the effectiveness of hierarchical frameworks in addressing these challenges [30].

In the second stage of hierarchical frameworks, **local correspondence methods** establish detailed relationships between features in the query and candidate images. **SuperGlue** enhances this process by modeling spatial context using a Graph Neural Network (GNN) [40], enabling it to identify consistent keypoint matches even under moderate viewpoint and illumination changes. It improves upon classical local matching methods by learning both feature descriptors and their spatial context. However, it still depends on the presence of clearly detectable keypoints, which may be sparse or unstable in low-texture settings.

To address the limitations of sparse keypoints, recent methods have shifted toward transformer-based architectures capable of computing dense pixel-wise correspondences. LoFTR uses a detector-free pipeline that combines CNN-extracted features with self- and cross-attention mechanisms, allowing it to compute dense, pixel-level correspondences [8]. This makes it highly suitable for low-texture or repetitive terrains, where classical methods often fail. LoFTR has outperformed SuperGlue on the MegaDepth dataset, which

includes significant viewpoint variation and repetitive patterns [56]. With an inference time of approximately 116 ms for 640×480 image pairs, it is suitable for real-time applications. However, its performance on desert-like or extraterrestrial landscapes remains an open research question.

Overall, the place matching stage plays a pivotal role in the success of loop closure detection. While global descriptor matching offers speed and scalability, local correspondence methods are more effective in visually challenging or ambiguous environments, where fine-grained spatial detail is required. A widely adopted strategy that combines the strengths of both is **hierarchical localization**. This approach first performs fast, coarse retrieval using global descriptors to narrow down the search space, and then applies local matching techniques to refine and verify the candidate matches. This two-stage process ensures computational efficiency and geometrical consistency.

2.2.5 Verification and Re-ranking

In challenging environments—especially those characterized by perceptual aliasing, repetitive structures, or sparse features—initial place recognition results may include false positives. To reduce these errors, many VPR systems incorporate an optional verification and re-ranking stage. This stage refines the shortlist of candidate matches by applying additional spatial or sensor-based checks to ensure geometric and physical plausibility before a loop closure is confirmed. Below, we review two key strategies used in this stage: spatial consistency verification and multi-modal fusion.

a) Spatial Consistency Verification

Spatial consistency verification methods assess whether the geometric relationship between a query image and a retrieved candidate is physically plausible. These techniques use local or dense feature correspondences to estimate the spatial transformation between two viewpoints, thereby validating the match based on geometric criteria.

A commonly used approach is **RANSAC-based** geometric verification [57], in which keypoint correspondences (often generated by methods such as ORB or SuperGlue) are used to estimate a fundamental or essential matrix. This estimation allows outlier correspondences to be discarded and ensures that the transformation between views is consistent with epipolar geometry. Similarly, **PnP (Perspective-n-Point)** algorithms combined with RANSAC can be used when 3D

landmarks are available, allowing for relative pose estimation based on 2D-3D correspondences [15, 16].

Another technique that integrates spatial validation into the matching process is patch alignment, as implemented in Patch-NetVLAD. This method evaluates the geometric consistency of retrieved matches by analyzing the relative vertical and horizontal displacement of corresponding image patches. Known as **Rapid Spatial Scoring** [47], this technique allows for efficient and robust verification of candidate images, especially in environments with distant or large-scale features, such as planetary landscapes.

Overall, these spatial verification methods significantly reduce the rate of false positives by filtering out candidate matches that are not geometrically consistent, thereby improving the reliability of loop closure detection.

b) Multi-Modal Fusion

In environments where visual data alone may be unreliable—such as those with poor lighting, dust, or feature repetition—verification can be improved through multi-modal fusion. These approaches combine image-based recognition with other sensor modalities to cross-validate matches and increase robustness.

One common strategy is **Visual-Inertial Odometry (VIO)** [58], which integrates camera data with inertial measurements from an Inertial Measurement Unit (IMU). The IMU provides high-frequency motion information that helps constrain visual matching and reduce drift, particularly in texture-poor regions or during rapid motion.

Another powerful approach is **visual–LiDAR** fusion [11], where point cloud maps generated by LiDAR are used alongside visual descriptors to verify place matches. This cross-modal validation is particularly useful in planetary robotics, where visual scenes may appear similar despite physical differences. By aligning image-based and LiDAR-based localizations, the system can reject perceptual aliases and reinforce only the matches that are consistent across both modalities.

These multi-sensor strategies are frequently integrated into SLAM frameworks, especially in robotics applications requiring high reliability. By combining different sensor modalities, they enhance robustness against perceptual ambiguity and improve the accuracy of loop closure decisions in visually degraded or repetitive environments.

2.3 Pose Estimation Techniques

Pose estimation refers to the process of determining the position and orientation (pose) of a camera or sensor relative to a known or reconstructed environment. In visual systems, this typically involves estimating a 6-degree-of-freedom (6-DoF) pose: three values for translation (x, y, z) and three for rotation (pitch, yaw, roll). Pose estimation can be performed using a single image (absolute pose) or by analyzing multiple views (relative pose). Depending on the available data—such as 2D images, depth maps, or 3D landmarks, various techniques can be applied, ranging from geometric solvers to deep learning models. These approaches vary in accuracy, robustness, and computational demands, especially when deployed in constrained or visually ambiguous environments such as planetary terrains.

2.3.1 Geometric Approaches

Classical pose estimation pipelines rely on geometric correspondences between 2D image features and known 3D points, or between image pairs, to compute the camera's position and orientation. These methods are valued for their accuracy and interpretability, but their performance is highly dependent on the quality of detected features and the robustness of matching under varying visual conditions.

A widely used technique in this category is the **Perspective-n-Point (PnP)** algorithm [15], which estimates camera pose from a set of 2D–3D correspondences, typically derived from keypoint matches or projected 3D models. To ensure robustness against outliers, PnP is often combined with **RANSAC** [59], which iteratively selects minimal subsets of correspondences to identify a geometrically consistent solution. This PnP-RANSAC combination remains a reliable baseline in structure-based localization tasks and is also a core component in hybrid pipelines such as COLMAP and FoundPose [17, 60].

For applications involving dense 3D data, such as from LiDAR or depth sensors, **Iterative Closest Point (ICP)** is commonly used to align two point clouds or depth maps and estimate relative pose [18]. ICP works by minimizing the distance between corresponding points in successive scans. While highly effective in well-structured environments, its accuracy can degrade significantly in scenes with poor geometric features, low texture, or poor initial alignment—conditions.

2.3.2 Deep Learning-Based Pose Estimation

Deep learning methods aim to directly regress the relative or absolute pose from images, bypassing traditional feature matching or explicit 3D scene geometry. These approaches can be broadly categorized into end-to-end pose regression, scene coordinate regression, and foundation model-based estimators.

a) End-to-End Relative Pose Regression

One of the earliest and most influential models in this domain is **PoseNet**, introduced by Kendall et al [14]. PoseNet uses a convolutional neural network to regress the absolute 6-DoF camera pose directly from a single RGB image. It offers high robustness to challenging visual conditions such as motion blur and lighting changes. However, PoseNet's performance is generally lower than that of geometry-based methods, particularly in terms of accuracy and metric scale estimation, due to its limited geometric supervision.

Building upon the foundations of PoseNet, more recent deep learning models have shifted toward end-to-end **relative pose regression (RPR)**, which predicts the relative transformation between a pair of input images. Early RPR approaches also used **convolutional backbones** such as ResNet-34 to generate global embeddings [61], followed by **MLPs** for pose regression. These models often lacked spatial awareness and struggled to generalize to unseen environments.

To address these limitations, more recent models have incorporated pretrained matching networks such as **LoFTR** to obtain spatially informed, semi-dense feature maps [62]. These are subsequently warped and passed through a camera motion regression module trained with specialized losses, such as cosine similarity for translation direction and L1 distance for scale, to decouple geometric components. This design improves generalization across datasets and offers faster inference than full feature matching pipelines, making it more practical for real-time applications.

A recent hybrid approach, **Match-And-Transform**, also known as Reloc3r, proposes a two-stage architecture that combines coarse feature matching with Transformer-based pose regression [63]. The model first identifies tentative correspondences using an attention mechanism inspired by SuperGlue, followed by a refinement stage that regresses the relative pose. Despite incorporating soft correspondences, the model's

interpretability remains limited since you cannot trace which image regions caused which part of the pose prediction. This makes debugging and safety validation harder than with classic matching + PnP pipelines, raising questions about the model's reliability and transparency in real-world or safety-critical deployments.

Another class of methods leverages **ViTs** for pose regression. By encoding positional information directly into patch embeddings, transformer-based models can implicitly emulate geometric algorithms like the **Eight-Point Algorithm** [64]. For instance, Rockwell et al. propose a lightweight architecture that modifies a standard ViT with an Essential Matrix Module (EMM), introducing bilinear attention, quadratic positional encodings, and a dual-softmax mechanism. These changes allow the ViT to approximate the Eight-Point Algorithm's key computation $\mathbf{U}^\top\mathbf{U}$ and estimate both rotation and translation with scale directly from image pairs. This approach achieves competitive accuracy on baseline datasets like Matterport3D [65], while requiring less data and outperforming CNN-based regressors in data-scarce settings.

b) Scene Coordinate Regression

This method offers an alternative by predicting, for each image pixel, its corresponding 3D coordinate in the world. This allows the camera pose to be estimated via **PnP** and **RANSAC**, bypassing explicit descriptor matching. A representative example is **Differentiable RANSAC (DSAC)** [59], which addresses a core limitation in integrating RANSAC into deep learning pipelines. Traditional RANSAC involves non-differentiable hypothesis selection, which makes end-to-end training unfeasible. DSAC resolves this by introducing a probabilistic formulation of hypothesis selection inspired by reinforcement learning, allowing the expected pose loss to be differentiated with respect to all network parameters. Applied to camera localization, DSAC enables end-to-end learning of scene coordinate predictions by directly optimizing pose accuracy, improving over classical methods. While later extensions introduce multi-view consistency and reprojection losses to improve robustness, these models often face scalability issues due to dense prediction requirements and high memory usage.

c) Foundation Model-Based Methods

Finally, foundation model-based methods have emerged as a new direction in pose estimation. **FoundPose**, for instance, leverages DINOv2

patch descriptors to establish 2D–3D correspondences between input images and synthetic renderings of object templates [60]. Using PnP with RANSAC and a featuremetric refinement stage, FoundPose accurately estimates 6-DoF object poses, even for symmetric or low-texture geometries. Notably, it does so without requiring task-specific training, demonstrating the potential of vision foundation models to support generalizable, training-free localization pipelines.

2.3.3 Hybrid Approaches

Hybrid methods combine the strengths of geometric and learning-based approaches to enhance robustness, scalability, and generalization in pose estimation tasks. Their flexibility makes them particularly attractive for applications such as long-term SLAM and planetary robotics, where environmental conditions can vary drastically and model adaptability is essential.

A well-established example is **SfM**, with pipelines such as **COLMAP** that reconstruct 3D models from collections of images through geometric triangulation and bundle adjustment [17]. For localization, new query images are matched against the reconstructed model using hierarchical or direct feature matching to estimate pose. Although COLMAP offers high accuracy and is widely used in research, its computational cost makes it impractical for real-time deployment.

Other approaches focus on efficient 2D–3D matching using learned descriptors. For example, **FoundPose** leverages synthetic template rendering, patch-level matching with DINOv2 descriptors, and pose estimation through PnP to localize objects without requiring per-scene training [60]. This results in a robust and training-free pipeline that generalizes well across various object types and visual conditions.

In more recent developments, differentiable matching pipelines integrate deep feature extractors, semi-dense matchers like LoFTR, and learnable pose regressors into a unified framework [62]. These setups leverage geometric priors while benefiting from data-driven learning and achieve competitive accuracy at lower inference times compared to full geometric methods.

2.4 Related work area

This section reviews key advances in multi-modal place recognition and pose estimation, with a focus on transformer-based architectures for both image and

point cloud data, as well as recent efforts in 3D point descriptor matching for LiDAR registration. We highlight relevant gaps that motivate the present work

2.4.1 Multi-Modal Place Recognition

Earlier LiDAR-only approaches, such as **PointNetVLAD**, aimed to directly learn global place descriptors from raw point clouds using PointNet and a NetVLAD aggregation layer [66, 12, 9]. These models demonstrated strong performance in large-scale place recognition without relying on images or precomputed features. However, their reliance on geometry alone limited their robustness in environments with perceptual aliasing or sparse structure. This limitation, along with the growing availability of synchronized visual and LiDAR data, motivated the development of multimodal approaches that can integrate complementary cues for more reliable localization under diverse conditions. A recent survey by Nagrani et al. provides a comprehensive review of such multi-modal place recognition methods, discussing fusion strategies, benchmark datasets, and remaining challenges including modality imbalance, scalability, and generalization across environments [67].

Recent research has increasingly focused on combining vision and LiDAR to leverage the complementary strengths of both modalities for place recognition in challenging environments. **AdaFusion** proposes an adaptive weighting mechanism that dynamically adjusts the contribution of visual and LiDAR features depending on the environment [13]. This is achieved via a dual-branch architecture—one for feature extraction and one for adaptive weighting—enhanced by multi-scale and inter-modality attention mechanisms. AdaFusion demonstrates robust performance across diverse scenes, outperforming traditional fusion methods that treat both modalities equally regardless of context.

Similarly, the **Unifying Local and Global Multi-modal Features (UMF)** framework introduces an attention-based fusion of image and LiDAR data using parallel branches with a ResNet-50 and a LiDAR encoder [11]. It incorporates both local and global descriptors through transformers with positional encoding and enhances final retrieval accuracy through re-ranking using geometric verification of local features. This method captures both coarse and fine-grained spatial features, boosting robustness against viewpoint and appearance changes.

Another notable contribution is **MinkLoc++** [68], which introduces a late fusion approach that processes RGB and LiDAR data separately and merges them into a global descriptor at the end of the pipeline. The

method employs deep metric learning with a triplet loss and addresses a key challenge in multimodal fusion: the *dominating modality problem*, where one modality (typically RGB) may overfit and degrade generalization. MinkLoc++ mitigates this by introducing a multi-head loss function with separate losses for each modality in addition to the fused descriptor. The architecture builds on MinkLoc3D with enhancements such as sparse voxel-based 3D convolutions and Efficient Channel Attention (ECA) to boost the quality of the point cloud representation. Experimental results on Oxford RobotCar and KITTI datasets demonstrate state-of-the-art performance [69, 70], particularly under challenging conditions like low visibility, showing that careful training strategies and robust architecture design are essential for effective multimodal place recognition.

2.4.2 Transformer-Based Feature Extraction

a) Vision Transformers

Recent advances in transformer-based feature extraction include the vision transformer **DINOv2**, which excels at learning structured patch-level representations that generalize across domains [34], unlike traditional CNN. Its robustness under severe viewpoint or appearance changes has made it particularly effective for VPR. Extensions such as **SALAD** further enhance DINOv2 by incorporating a feature aggregator based on optimal transport [26], achieving state-of-the-art performance. The combination of patch-level descriptors with a global token makes this approach especially suitable for place recognition under challenging visual conditions.

Additionally, transformer-based self-supervised learning approaches like **Masked Autoencoder (MAE)** and hierarchical attention structures such as **ASpanFormer** have been explored for learning generalizable image representations [71, 72]. These models, while not specific to planetary exploration, offer promise by avoiding the need for task-specific labeled data.

b) Point Cloud Transformers Point Cloud Transformers

In the domain of 3D data, **Sonata** stands out as a self-supervised transformer-based architecture designed for learning point cloud representations [73]. It achieves strong performance in various segmentation and matching tasks by overcoming limitations found in previous sparse convolutional or U-Net-based methods [74, 75,

76]. Sonata discards the decoder and focuses solely on encoder-side feature learning, enabling multi-scale spatial reasoning and semantic awareness, which are crucial for LiDAR-based place recognition and alignment.

2.4.3 3D Point Descriptor Matching

LiDAR scan registration for long-term localization often relies on reliable 3D point descriptors to establish correspondences between scans and maps. Traditional methods like **Fast Point Feature Histograms (FPFH)** or **3DMatch** are increasingly replaced by learning-based approaches [77, 78]. A recent and novel direction is the use of visual foundation models to guide 3D point descriptor learning.

In particular, the method described in *LiDAR Registration with Foundation Models* utilizes DINOv2 to extract dense 2D image features [79], which are projected onto the corresponding 3D point cloud to serve as descriptors. These descriptors are then used in conjunction with traditional geometric registration methods like **RANSAC** and **ICP** [16, 18]. This hybrid approach outperforms many learning-based baselines, particularly in long-term map registration tasks with substantial environmental changes. Its generalization capability and independence from LiDAR-specific network training make it attractive for planetary applications where domain shifts are common.

Chapter 3

Methodology: A Multi-Modal Hybrid Approach for Visual-LiDAR-Based Localization

This chapter presents the methodology underlying the development of a multi-modal system for place recognition and pose estimation in unstructured environments. The approach is rooted in applied experimental engineering, combining deep learning-based feature extraction with classical geometric methods to support loop closure in SLAM systems. The methodology is shaped by the challenges of planetary-like terrains, such as low texture, limited geometric structure, and perceptual aliasing, and reflects the need for robustness and computational efficiency in real-time robotic applications.

Section 3.1 introduces the system architecture and justifies the selection of models and techniques used for visual representation, multi-modal fusion, and geometric verification. Section 3.2 describes the dataset and data preparation process, including the definition of ground truth loop closures and the overlap scoring function. Section 3.3 outlines the experimental design and benchmarking protocol, covering task separation, evaluation metrics, baseline comparisons, and validation procedures.

Together, these sections provide a transparent and technically grounded framework for evaluating loop closure performance under realistic and demanding conditions.

3.1 System Architecture and Methodological Rationale

3.1.1 Hybrid and Multi-Modal Approach

The system presented in this work follows a hybrid, multi-modal design that integrates deep learning techniques with classical geometric pose estimation methods. This combination is motivated by the need to handle the unique challenges of planetary-like environments while ensuring the system remains suitable for real-time operation. The environments under consideration, such as Martian analog sites or volcanic terrains, are characterized by weak textures, repeated patterns, and sparse geometric features. These conditions pose significant difficulties for both traditional and deep learning-based approaches when used in isolation.

At the core of the system lies a hierarchical architecture inspired by previous work in hierarchical localization [10]. This architecture enables a multi-stage place recognition and pose estimation pipeline, balancing computational cost with accuracy. The design begins with a fast, coarse filtering stage that uses global visual descriptors generated by the SALAD network [26]. These descriptors, of size 8192, are compared using cosine similarity and FAISS indexing to rapidly retrieve the top 20 candidate images from the database [27]. This stage ensures fast and efficient screening, eliminating the need for expensive computations on every frame.

Once this shortlist is obtained, a second, finer screening process is carried out. Features are extracted from the final three layers of the DINOv2 transformer model for both the query image and the top 20 candidates [34]. These features offer more detailed and semantically rich representations, which are then compared to produce a refined list of the top 10 most relevant candidates. Following these two filtering stages, the system performs feature matching and relative pose estimation only on this smaller set of candidates. Since these geometric computations are applied selectively to a limited number of candidates, the process remains efficient without sacrificing accuracy. The final output is a ranked list of candidates reordered based on their estimated spatial relation to the query frame. This multi-stage architecture is visualized in Figure 3.1, which illustrates how the different components—global retrieval, local feature matching, and geometric verification—interact to support robust localization in challenging environments.

Alternative approaches that attempt to jointly solve place recognition and

pose estimation, such as BoxGraph [80], were intentionally avoided in this system. While these methods offer integrated solutions, they come with significant computational overhead, making them unsuitable for real-time applications on resource-limited platforms. Moreover, BoxGraph and similar methods typically require a well-structured 3D point cloud map, often derived from urban LiDAR scans, along with precomputed semantic segmentations of the environment. These segmentations extract high-level features like object type, centroid position, and bounding boxes, which are useful in structured urban scenes. However, in planetary-like environments, where distinct objects are scarce and the landscape lacks semantic richness, such preprocessing would add complexity without a corresponding gain in performance.

By adopting a hierarchical structure, the system maintains modularity and supports scalability. Each component—coarse retrieval, fine descriptor comparison, and final pose verification—can be independently modified or optimized for different datasets, environments, or computational requirements. This modularity also allows for runtime adaptations, such as dynamically adjusting the number of candidates retrieved (top-k) or tuning the precision of pose estimation based on available resources.

The decision to combine deep learning methods with classical geometric techniques also reflects a deliberate trade-off between efficiency and robustness. Deep learning models like DINOv2 are capable of extracting high-level features that remain informative even in texture-less or homogeneous environments [34]. However, they are computationally intensive and typically require more memory and processing time. In contrast, geometric methods such as RANSAC are fast and interpretable but depend on reliable keypoint detection and matching [16]. By using deep learning to extract features and match keypoints, and then applying geometric methods for the final pose verification, the system effectively combines the strengths of both paradigms.

A purely classical pipeline—using feature extractors like SURF or ORB combined with geometric pose estimation—would not be sufficient for the target environments [29, 6]. These classical methods are designed to detect regions with high intensity variation such as corners or edges. In desertic or planetary terrains, where images are often flat and texture-poor, these methods fail to detect enough reliable keypoints, leading to degraded recognition performance. On the other hand, a fully deep learning-based pipeline would require more computational resources and would not benefit from the spatial interpretability and robustness of geometric verification. Additionally, such models typically lack guarantees about spatial consistency, which is especially important in SLAM and localization tasks.

The inclusion of both visual and LiDAR data further strengthens the system's adaptability to harsh and unstructured environments. Visual inputs offer rich appearance-based information that can be leveraged through powerful deep models. However, in settings where visual cues are weak—due to lighting conditions, surface homogeneity, or repetitive structures—vision alone may be unreliable. To address this, the system incorporates LiDAR scans as a complementary modality. LiDAR contributes dense geometric information that remains consistent under variable lighting, providing valuable structural context.

Purely LiDAR-based systems, such as "One RING to Rule Them All" [81], have also been explored for joint place recognition and pose estimation. While effective in some cases, these systems often rely on prebuilt 3D maps and typically convert LiDAR scans into 2D Bird's-eye View (BeV) representations [82]. These projections are then processed using techniques such as the Radon Transform [83]. However, such projections inherently lose critical 3D information, which reduces the system's ability to discriminate between similar-looking scenes, particularly problematic in terrains with repetitive structures or limited variation, as commonly found in planetary analog environments.

The importance of combining modalities has been further demonstrated in recent work such as UMF [11]. This study shows that integrating LiDAR and image data significantly boosts performance in place recognition tasks under challenging conditions. While LiDAR point clouds alone may be too sparse or flat to support accurate candidate retrieval, they become highly useful in the final verification stage—especially when nearby geometric structures, like rocks or terrain features, are present. These structures allow for more reliable point correspondences and pose estimation, which helps disambiguate visually similar locations.

In the proposed system, both images and LiDAR scans are utilized. Global visual descriptors drive the initial retrieval stages, ensuring speed and scalability. In the later stages, local multi-modal features—combining visual and LiDAR information—are used to verify candidates and compute accurate relative poses. This combination allows the system to remain effective across a wide range of scenarios, including those where one modality alone would fail.

In conclusion, the design choices made in this system are driven by the constraints and demands of real-world deployment in planetary-like environments. By adopting a hierarchical structure, combining deep learning with geometric methods, and incorporating multiple sensing modalities, the

system achieves a balance between precision, efficiency, and adaptability. It supports generalization across different terrains and remains computationally feasible for use in real-time SLAM systems operating on resource-constrained robotic platforms.

3.1.2 Visual Feature Extraction and Aggregation

Visual place recognition in unstructured environments requires feature representations that are robust to changes in viewpoint, lighting, and scene composition, especially in terrains with limited texture or distinct visual landmarks. For this reason, transformer-based image encoders have become increasingly favored over convolutional backbones due to their ability to capture high-level semantic and structural cues across the entire image [30]. Among these, DINOv2 has demonstrated state-of-the-art performance for unsupervised representation learning and has been adopted in this work as the core image feature extractor [34, 38].

The base variant of DINOv2 was selected as it offers an optimal balance between performance and computational cost. Empirical results from existing benchmarks show that while the large and giant variants produce marginally better retrieval accuracy, they introduce substantial memory overhead and latency [34]. Conversely, the small variant underperforms significantly in scenarios with weak textures and spatial ambiguity, failing to extract meaningful descriptors. The base model, with a 768-dimensional output and 86M parameters, offers a strong trade-off between feature richness and efficiency, making it a suitable backbone for real-time applications (see Table 3.1).

Table 3.1: Comparison of transformer-based models by size, parameters, latency, and R@1. Adapted from [26].

Model	**Dim. size**	**# Params.**	**Latency (ms)**	**R@1**
Small	384	21 M	1.30	90.5
Base	768	86 M	2.41	92.2
Large	1 024	300 M	7.82	92.6
Giant	1 536	1 100 M	24.93	91.7

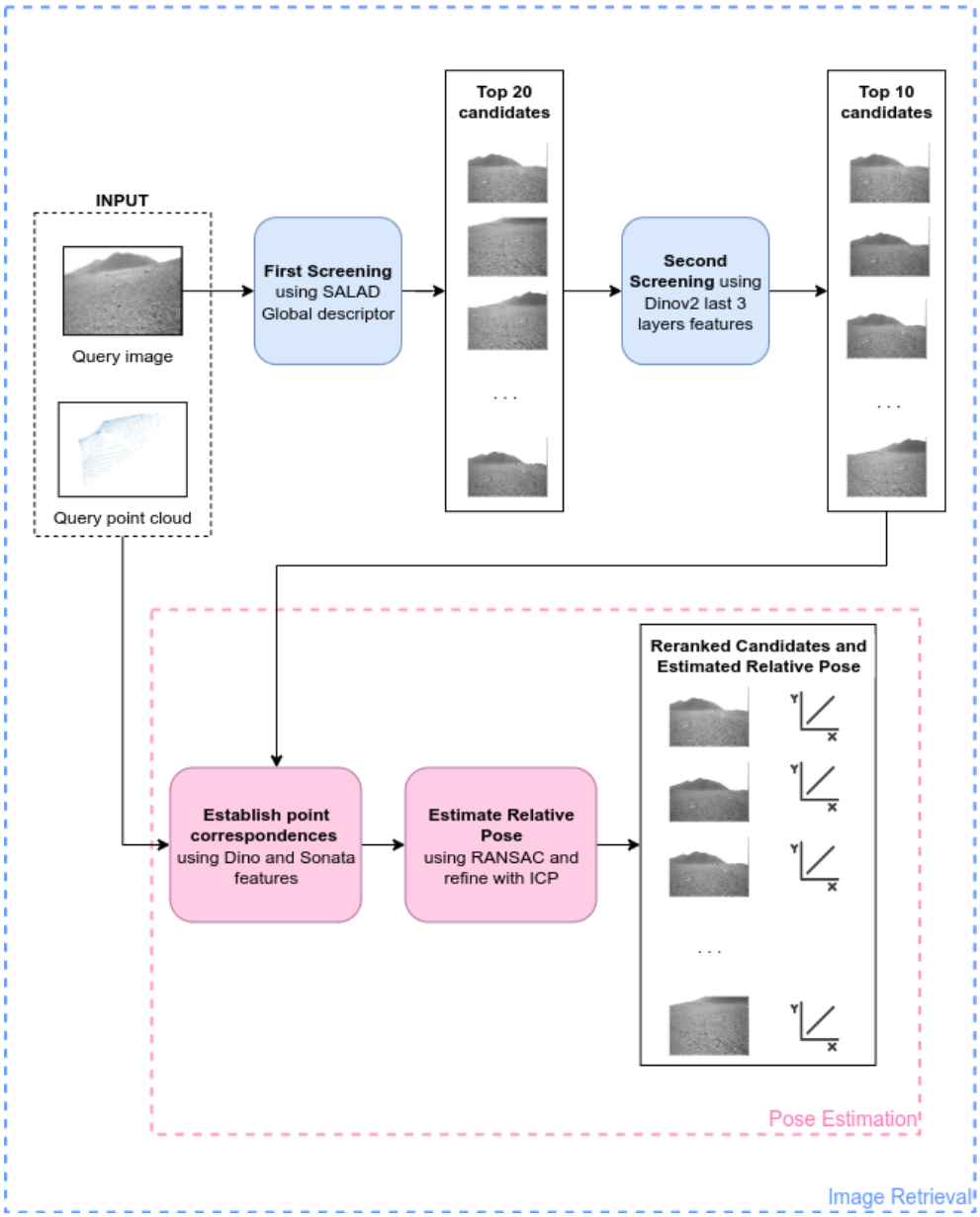


Figure 3.1: Overview of the proposed hybrid multi-modal localization pipeline. The system combines global visual retrieval using SALAD descriptors, fine-grained DINOv2 feature matching, and geometric pose estimation with LiDAR data. A hierarchical architecture enables efficient candidate filtering and accurate pose estimation, making it suitable for deployment in environments with weak textures and repetitive structures.

As shown in Step 1 of the pipeline in Figure 3.1, the system integrates the SALAD network as an aggregation module to generate global descriptors from image features. SALAD builds upon the NetVLAD architecture by introducing several key improvements that directly address the limitations of its predecessor. Unlike NetVLAD, which initializes its clustering weights with k-means centroids and applies softmax-based soft assignment [9], SALAD treats the assignment step as an optimal transport problem [26]. It includes a "dustbin" cluster to discard uninformative features, and uses the Sinkhorn algorithm to compute the optimal assignment of image patches to clusters. This mechanism allows SALAD to focus on salient image regions and avoid incorporating noisy background features, which is particularly advantageous in the low-structure scenes.

Moreover, while NetVLAD computes residuals based on centroids and aggregates them per cluster, SALAD directly processes features through two fully connected layers, followed by a sum aggregation and a concatenation with a global token descriptor. This enables more efficient dimensionality reduction and yields a compact, 8192-dimensional descriptor that captures both global context and localized structure. Another advantage of SALAD is its compatibility with DINOv2 features: unlike simpler pooling methods such as GeM [44, 45], SALAD effectively exploits the positional and semantic richness of transformer patch embeddings. These properties make it a compelling choice for generating descriptors that remain discriminative under extreme viewpoint or illumination variations.

For efficient candidates search, the system employs FAISS with an IndexFlatIP index and L2-normalized descriptors [27]. This setup allows cosine similarity to be used as the scoring metric, enabling fast and accurate retrieval at scale. From the database, the top 20 most similar candidates to a query image are retrieved using their global SALAD descriptors. This coarse retrieval stage is designed to be lightweight and scalable, enabling rapid screening of large image sets while maintaining a high probability of retrieving true positive matches. The two-stage image retrieval process is illustrated in Figure 3.3.

The use of cosine similarity, combined with transformer-based descriptors, offers robustness to intensity variations and affine transformations, which are common in outdoor environments. Cosine similarity is widely used in place recognition pipelines due to its effectiveness in comparing high-dimensional feature vectors, especially when descriptor magnitudes may vary across inputs [84, 85, 86].

For computational efficiency, all global and patch-level image descriptors

produced by DINOv2 and SALAD are precomputed and stored in serialized .pickle files. This allows for rapid retrieval during evaluation and avoids redundant forward passes at runtime.

Alternative approaches such as NetVLAD or GeM were considered, but were ultimately rejected due to either lower performance (in the case of GeM) or higher descriptor dimensionality and sensitivity to initial clustering (in the case of NetVLAD) [26]. NetVLAD, for instance, requires careful initialization and suffers from soft assignment limitations in ambiguous scenes. In contrast, SALAD offers an end-to-end trainable aggregation method with better generalization across varied environments.

In summary, the combination of **DINOv2 (base) for image encoding**, **SALAD for descriptor aggregation**, and **FAISS with cosine similarity for efficient retrieval** forms a retrieval pipeline that is both accurate and computationally tractable. This visual backbone enables robust candidate selection even under challenging conditions, laying a solid foundation for the subsequent pose estimation stages of the system.

3.1.3 Geometric Verification and Pose Estimation

Once a shortlist of candidate frames has been retrieved using visual descriptors, the system proceeds to verify the candidates and estimate their relative pose with respect to the query. This stage is critical for closing loops in SLAM, as it determines the actual geometric consistency between views beyond visual similarity. For this purpose, the system integrates point cloud-based geometric verification using LiDAR data, enhanced with visual information from the DINOv2 encoder. The final pose is estimated through a robust combination of descriptor matching, RANSAC-based alignment, and ICP refinement, as illustrated in Figure 3.3.

The method relies on two key components: **Sonata for LiDAR-based feature extraction** and **DINOv2 for image-based descriptors**. Sonata was chosen as the 3D encoder due to its ability to extract patch-level features from unstructured point clouds [73]. It divides each LiDAR scan into spatial patches of fixed size (e.g., 8×8×8 meters) and outputs a 512-dimensional embedding per patch. Unlike handcrafted descriptors or handcrafted segmentation, Sonata is pretrained on large-scale 3D data and is capable of producing semantically meaningful and robust features, even in sparse or noisy point clouds. While fine-tuning Sonata on the target domain was initially considered, it proved infeasible due to hardware limitations. The training was attempted on a Slurm cluster equipped with 2× Quadro GV100

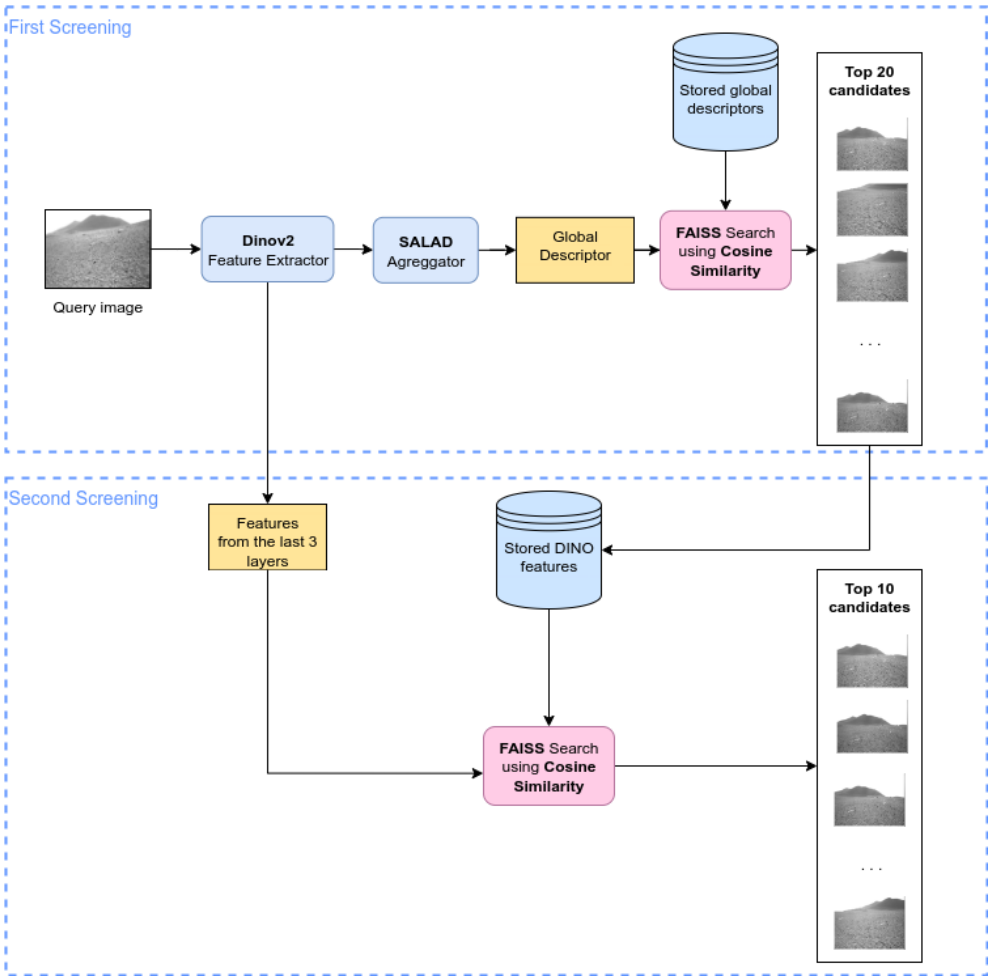


Figure 3.2: Two initial stages of the image retrieval pipeline for place recognition. **First Screening:** Features are extracted from the query image using DINOv2 and aggregated into a global descriptor via the SALAD module. This descriptor is matched against a database of stored global descriptors using FAISS with cosine similarity, retrieving the top 20 candidates. **Second Screening:** For a finer selection, features from the last three layers of DINOv2 are used and compared against stored features, again using FAISS with cosine similarity, to produce a final shortlist of the top 10 candidates. The integration of transformer-based features, SALAD aggregation, and cosine similarity provides robustness to environmental variations and visual ambiguities.

GPUs, 128 GB of RAM, and an 18-core Xeon CPU. However, persistent out-of-memory (OOM) errors occurred early during training, even with modest batch sizes. In contrast, the original Sonata model was trained using a

distributed setup with a batch size of 96 across 32 GPUs, allowing significantly higher memory throughput and parallelism. Given these constraints, the pretrained Sonata model was used in this work. Despite being trained on a general-purpose dataset, the pretrained version still offered stable and meaningful features suitable for downstream matching.

To enable multi-modal feature integration, point-level features from DINOv2 are projected into 3D space using camera intrinsics and frame calibration. Inspired by recent work in LiDAR-vision registration [79], this projection enables each 3D point to inherit visual descriptors from the corresponding image patch. The visual patch descriptors are taken from the last three layers of the DINOv2 transformer and concatenated to form a 2304-dimensional vector. These are then associated with each 3D point based on its projection into the image plane. The resulting multi-modal descriptor is formed by concatenating the DINO-based patch embedding with the corresponding Sonata feature, creating a fused feature representation that encodes both visual appearance and geometric structure.

Matching between the query and candidate frames is performed by comparing these multi-modal point descriptors using cosine similarity, which was found to be more effective than Euclidean distance or Lowe's ratio in this setting. A simple similarity threshold is applied to filter reliable correspondences, avoiding the need for more complex mutual matching algorithms. This point-level filtering ensures that only high-confidence matches are passed to the geometric estimation stage.

The initial pose estimation is conducted using the **RANSAC** algorithm [16], which robustly fits a rigid SE(3) transformation (Special Euclidean group in 3D—between the matched 3D point sets). SE(3) models real-world rigid-body motion by combining rotation and translation while preserving distances and angles. It excludes transformations such as scaling or shearing, making it the standard for estimating 6-DoF poses in SLAM and 3D vision tasks. The Open3D library is used to perform this step, leveraging its implementation of point-to-point transformation estimation with a configurable inlier distance threshold [87]. RANSAC is chosen for its ability to handle noisy matches and outliers [16], which are expected in environments where geometric structure is often sparse or ambiguous. The estimated transformation provides an initial guess of the relative pose between the query and each candidate.

Following RANSAC, the pose is refined using the **ICP** algorithm [18]. A point-to-point ICP variant is applied to the same 3D point sets [88], using the RANSAC pose estimation as an initialization. This step further improves pose accuracy by minimizing the Euclidean distance between corresponding points.

Although more advanced ICP variants such as point-to-plane or color ICP exist [88, 89], the selected point-to-point approach offers sufficient accuracy for the application domain without introducing unnecessary complexity. In particular, color ICP was not applicable in this case due to the lack of reliable color information in the LiDAR point clouds, and point-to-plane ICP requires accurate surface normal estimation, which is challenging given the sparsity and noise in these data.

In addition to estimating the relative transformation, the system uses the ICP fitness score to determine whether a candidate should be retained or discarded. The fitness score, defined as the proportion of inlier correspondences relative to the total number of points, provides a quantitative measure of geometric consistency. Candidates with low fitness values are filtered out as unreliable matches. The final re-ranking of accepted candidates is performed based on their estimated spatial proximity to the query frame, ensuring that the top-ranked results are not only geometrically consistent but also contextually relevant within the trajectory.

Alternative verification methods, such as learned pose regression networks or matching modules like SuperGlue, were evaluated but ultimately not selected. Learned pose regression approaches tend to generalize poorly to novel environments and lack geometric interpretability in their predictions. Meanwhile, vision-only matchers such as SuperGlue or LoFTR depend heavily on sufficient texture and repeatable features, which are often absent in planetary terrains. In contrast, the proposed geometric pose estimation pipeline is better suited for loop closure in sparse, ambiguous, or low-visibility conditions.

While the proposed pipeline is robust and computationally efficient, a few limitations remain. The pretrained Sonata model, although stable, may not be optimally adapted to planetary terrain due to domain shift from urban datasets. Fine-tuning was not feasible under hardware constraints, but may improve performance in future work. Similarly, the projection of image features into 3D assumes accurate camera calibration and synchronized modalities; misalignment could degrade descriptor fusion quality. Lastly, while RANSAC and ICP are well-established, their performance depends on the quality of initial correspondences, and alternative robust estimators could be explored for further gains.

In summary, the geometric verification and pose estimation module combines robust 3D feature extraction, visual-geometric fusion, and classical alignment techniques to produce accurate and interpretable pose estimates. This design supports modularity and adaptability, making it well aligned with

the constraints and goals of SLAM systems operating in unstructured, real-world conditions.

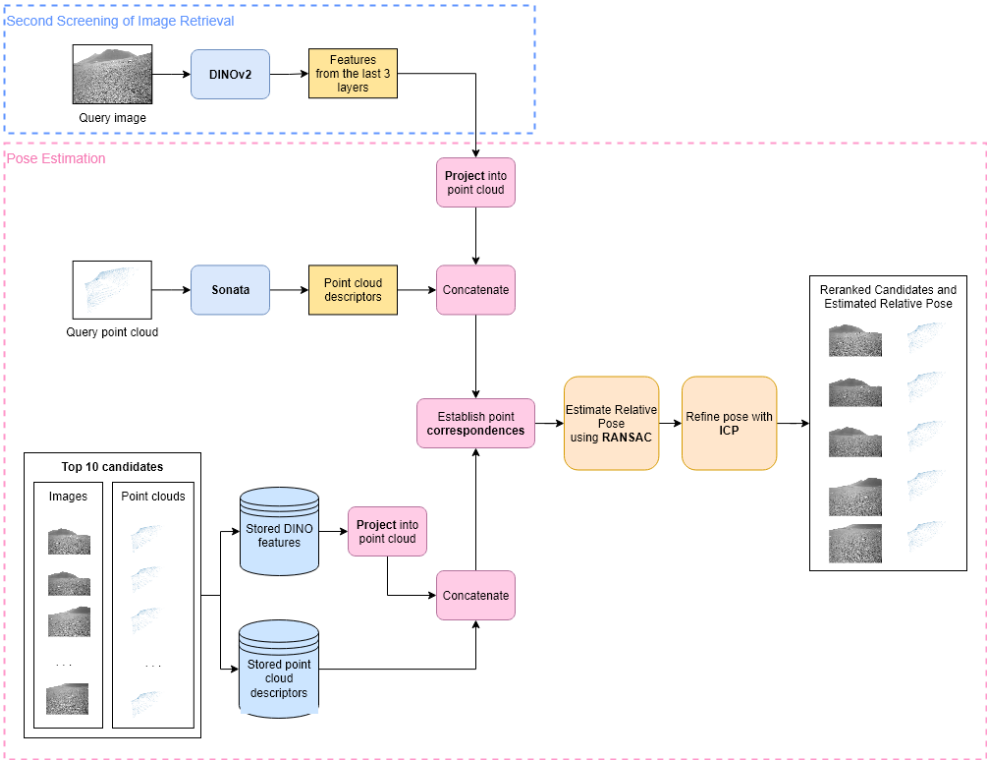


Figure 3.3: Overview of the **geometric verification** and **pose estimation** pipeline. Given a query image and point cloud, visual features are extracted from the last three layers of DINOv2 and projected into 3D space using known calibration. These are concatenated with LiDAR-based descriptors from Sonata to form fused point-level embeddings. For each of the top 10 retrieved candidates, stored DINO and Sonata descriptors are processed similarly. Point correspondences are then established by comparing fused descriptors, followed by pose estimation using RANSAC and refinement with ICP. The final output includes re-ranked candidates based on geometric consistency and the estimated relative pose.

3.2 Dataset and Data Preparation

This project uses the Etna dataset, a publicly available multi-modal dataset collected in a Moon-like environment on Mount Etna, Sicily [21]. It was

chosen specifically for its relevance to planetary robotics: it exposes extreme environmental challenges such as visual aliasing, sparse and ambiguous geometric structure, and lighting variability. These characteristics closely resemble conditions expected in extraterrestrial exploration missions and provide a rigorous testbed for developing robust loop closure techniques. The dataset includes grayscale images, LiDAR point clouds, precise camera intrinsics and extrinsics, robot poses (position and orientation relative to North), and synchronized timestamps. Figure 3.4 presents a representative example of a single frame from the Etna dataset, showing both the grayscale image and the corresponding LiDAR scan.

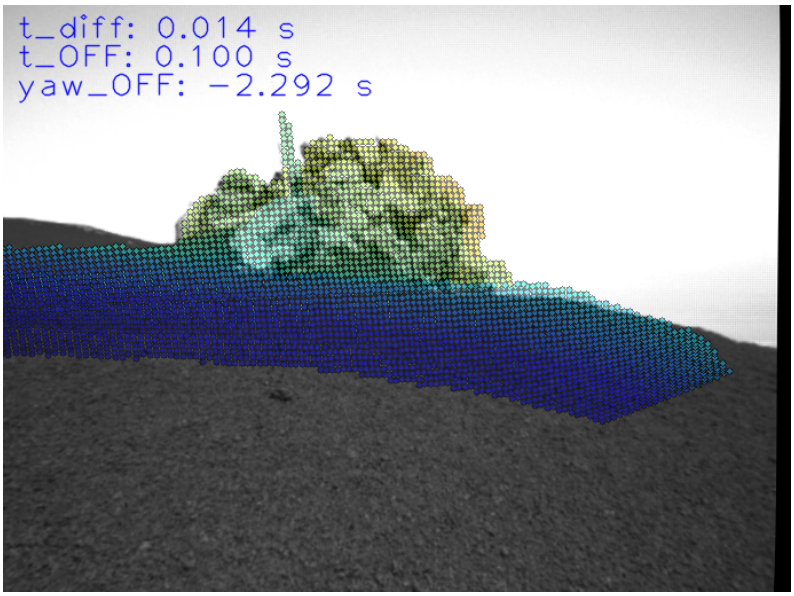


Figure 3.4: Example of a loop closure pair detected using the overlap function. Despite changes in viewpoint and lighting, the location is the same.

The dataset consists of seven sequences, each with different motion patterns and landscape characteristics. Figure 3.5 shows a top-down trajectory visualization of all sequences in the Etna dataset. While the figure does not explicitly annotate loop closures, it illustrates the diversity of the paths and the spatial layout of the terrain. For this work, two sequences—`s3li_loops` and `s3li_traverse1`—were used to generate query frames for evaluation. These sequences were selected because they contain multiple revisits to the same locations either within the sequence or relative to other sequences, enabling natural loop closures. The remaining five sequences (`crater`, `crater_inout`, `landmarks`, `traverse2`, and `mapping`) were used to

construct the database during fine-tuning. This split ensures that the system is evaluated on data with previously unseen loop closures.

Figure 3.5: Top-down view of the trajectories recorded in the Etna dataset. The plot shows the spatial paths of all seven sequences captured during data collection.

To define ground truth matches for loop closure detection and pose estimation, a custom viewpoint overlap function was developed (see Appendix A, A.1). Traditional pair selection strategies based solely on spatial distance or timestamp proximity often capture consecutive or near-consecutive frames, which are not representative of true loop closures. In this work, we are specifically interested in detecting revisits to the same location under different viewpoints, which is why a minimum timestamp separation of 100 ms is enforced to exclude sequential matches. Figure 3.6 illustrates an example of two frames from different timestamps that satisfy the loop closure criteria, showing significant viewpoint variation while capturing the same location.

Two alternative formulations for computing viewpoint overlap were considered. The first (`compute_overlap_v1`) estimates angular alignment by comparing the yaw angles of two frames and modulates the result with a position-based correction derived from their relative distances along the lateral and forward axes. The second variant (`compute_overlap_v2`) uses polygonal approximation to explicitly compute the intersection of each camera's field-of-view triangle, treating overlap as a normalized area

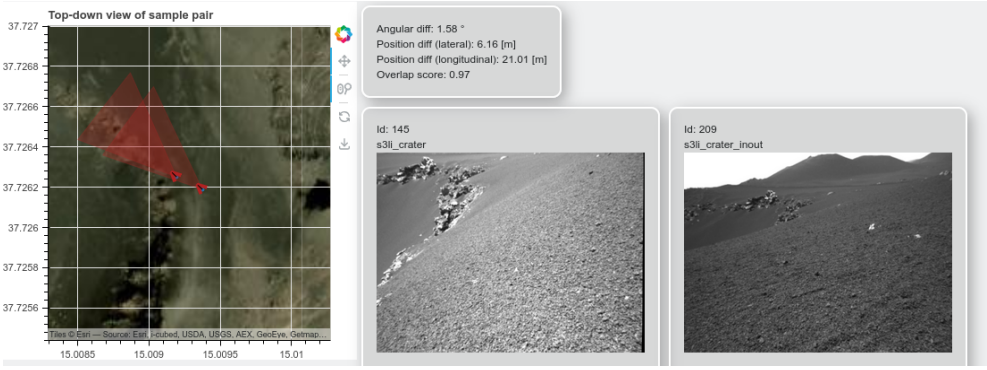


Figure 3.6: Example of a loop closure pair detected using the overlap function. Despite changes in viewpoint and lighting, the location is the same.

of intersection. While this method is geometrically intuitive, it proved less effective in practice due to sensitivity to Field of View (FOV) assumptions and reduced flexibility in controlling spatial constraints. As a result, `compute_overlap_v1` was selected for its better alignment with perceptual overlap and its tunability for the given environment and sensor configuration.

The final overlap score is computed as the product of angular consistency and spatial alignment factors. This continuous metric allows fine-grained control over what constitutes a ”true positive” match for loop closure.

Let $\tau_t = 100$ ms denote the minimum temporal separation and $\tau_o = 0.6$ the minimum overlap score. Image pairs with a timestamp difference greater than τ_t and an overlap score above τ_o were considered valid loop closure matches. These thresholds were empirically tuned to balance between strict geometric consistency and sufficient coverage of revisited scenes. The use of such a scoring function makes the evaluation procedure more reflective of real-world robotic constraints, where viewpoint overlap—not just proximity—is the key factor for reliable loop detection.

The Etna dataset is distributed under an open-source license and does not include any human subjects or privacy-sensitive data. It is publicly available online [21], along with detailed documentation and usage guidelines.. The sensor suite includes a stereo camera, a solid-state LiDAR, and an IMU, and the dataset provides accurate D-GNSS ground truth for all frames. The LiDAR’s narrow field of view (70° horizontal × 30° vertical), in combination with the environment’s lack of salient features, makes this dataset particularly challenging for both visual and LiDAR-based place recognition. These

characteristics support the methodological focus of this work on developing robust, multi-modal loop closure techniques in unstructured and visually ambiguous environments.

In addition to the Etna dataset used for evaluation and DINOv2 fine-tuning, the SALAD aggregation module was re-trained using a combination of standard visual place recognition datasets—the same ones used by the original authors. These include the Mapillary Street-Level Sequences (MSLS) dataset [23], the GSV Cities dataset [24], and the Pittsburgh250k dataset from the original NetVLAD benchmark [25]. All three datasets consist of large-scale urban imagery with GPS-based ground truth, making them well-suited for training global image descriptors using contrastive or triplet loss. The datasets were used directly without additional preprocessing or domain adaptation, and the training procedure followed the protocol described in the SALAD paper [26]. This training step was conducted to explore potential improvements in the aggregation module—within the image retrieval pipeline—by modifying the model's input dimensionality. Importantly, these datasets were used solely for training and were not involved in any aspect of evaluation, testing, or loop closure analysis.

3.3 Evaluation Design and Benchmarking Strategy

This section outlines the experimental design used to evaluate the two core components of the system—image retrieval and pose estimation—and describes the metrics, benchmarks, and validation strategies applied. The evaluation is performed on the Etna dataset, which presents a planetary-like environment characterized by sparse features and visual ambiguity, making it a rigorous testbed for localization systems operating in unstructured terrain.

3.3.1 Retrieval and Pose Estimation Benchmarks

The evaluation is structured around two distinct tasks: image retrieval and pose estimation. These components are assessed separately to isolate their performance and to better understand the contribution of specific model choices to each sub-task. Feature extraction and fusion are not directly evaluated as standalone modules but are instead reflected in the performance of the retrieval and pose estimation pipelines.

For **image retrieval**, the comparison includes a diverse set of baselines

categorized by modality and model type. **CNN-based** methods are represented by NetVLAD [9], a widely used architecture for place recognition that aggregates convolutional features into a global descriptor using a learned VLAD-based pooling scheme. **Transformer-based visual-only** methods include several variants of DINOv2 [34]: a small model using patch features from the last three layers with cosine similarity, a base model using the [CLS] token, and a base model using patch embeddings aggregated from the final three layers. The system proposed in this work also uses DINOv2 (base model) but integrates the SALAD module to produce global descriptors [26]. To isolate SALAD's performance, it is also evaluated independently as a retrieval method using cosine similarity on its standalone global descriptors. The benchmark additionally includes **multimodal** methods, such as MinkLoc++ [68], which combines LiDAR and visual inputs, and **LiDAR-only** methods, such as PointNetVLAD [66]. Two additional multimodal frameworks—UMF and AdaFusion—were initially considered for comparison [11, 13], but due to unresolved technical and availability issues, they could not be included in the final evaluation.

The **pose estimation** task is evaluated using a range of baselines covering different algorithmic strategies. **Handcrafted geometric** methods are represented by FPFH combined with RANSAC [77, 16]. **Transformer-based** pipelines include LoFTR for 2D keypoint matching [8], DINOv2 patch embeddings directly compared between frames [60], and a DINO+LiDAR fusion pipeline inspired by recent multi-modal registration research [79]. Another transformer-based baseline uses Sonata's 3D patch embeddings alone [73], matched with cosine similarity and refined with RANSAC. Finally, a **regression-based** method, Reloc3r [63], is included to test performance using an end-to-end image-based 6-DoF pose predictor. All methods were implemented to operate on the same Etna data split used for evaluating the proposed system, ensuring consistency in environmental conditions and test scenarios.

3.3.2 Evaluation Metrics

To evaluate image retrieval performance, the chosen metric is precision at top-k, reported as **Precision@1**, **@5**, and **@10**. This reflects the priority in SLAM applications for high-confidence loop closure predictions: false positives are far more damaging than occasional missed matches. According to the ground truth definition ($\tau_o = 0.6$, $\tau_t = 100$ ms), a retrieved image is considered a correct match if it meets these criteria, thereby excluding temporally adjacent

frames. This ensures that only genuine loop closures are counted, providing a more realistic assessment of the system's place recognition capability.

Pose estimation accuracy is measured using three main metrics: **yaw error** (in degrees), and **translation errors** along the x and y axes (in meters). These are computed by comparing the estimated relative transform with the ground truth relative pose. To facilitate interpretation, the analysis also includes cumulative accuracy plots, reporting the percentage of poses that fall within defined thresholds—2°, 3°, 5°, and 10° for yaw; and 1, 2, 3, 5, and 10 meters for both x and y translation.

Runtime is also measured for both components: image retrieval time, pose estimation time, and the total inference time per frame. All experiments are executed using custom Python scripts developed for this thesis.

3.3.3 Ensuring Validity and Reliability

Multiple strategies were employed to ensure the validity and reproducibility of the experimental results. Ground truth loop closures were determined using a custom scoring function (`compute_overlap_v1`, see Appendix A, A.1) that combines angular similarity and spatial consistency, offering a continuous and tunable overlap metric. Thresholds such as τ_o and τ_t were set to select valid matches and eliminate sequential frames, respectively. The correctness of the matching criteria was verified through visual inspection using the S3LI toolkit [90].

To ensure fair comparisons, all methods—both proposed and baseline—were tested on the same set of query frames, database entries, and retrieval settings. The same top-k candidate strategy was applied uniformly. While no cross-validation or random seeds were used, multiple runs were conducted, and no significant variability was observed. All experiments were fully scripted, and the evaluation code will be released to ensure full reproducibility and support future extensions by the research community.

Chapter 4

System Implementation and Technical Design

This chapter presents the complete implementation of the proposed system for multi-modal loop closure detection and pose estimation in unstructured environments. Section 4.1 details the data preparation procedures, including ground truth generation and training data construction. Section 4.2 introduces the feature extraction and aggregation pipeline, covering the use and fine-tuning of DINOv2 and its integration with the SALAD aggregation module. Section 4.3 describes the descriptor storage strategy and the retrieval infrastructure based on FAISS. Section 4.4 outlines the pose estimation pipeline, including 3D projection, multi-modal fusion, and geometric verification.

4.1 Dataset Preparation and Ground Truth Generation

To enable effective evaluation of loop closures in the Etna dataset, a custom data preparation pipeline was implemented. This included generating reliable ground truth correspondences between frames using a viewpoint overlap scoring function and refining data selection with the aid of visual verification tools. The resulting loop closure pairs and training triplets were saved in both `.csv` and `.pkl` formats for convenient inspection and efficient pipeline integration, respectively.

The overlap computation relied on the `compute_overlap_v1` function (see Appendix A, A.1), which estimates frame similarity based on angular

alignment (yaw comparison) and positional consistency (relative distances along the forward and lateral axes). Parameters such as the maximum lateral and longitudinal distances (set to 60 and 80 respectively) were empirically tuned to maximize alignment with perceptual overlap. This was validated using the visualization toolkit from the S3LI framework, which enables side-by-side inspection of matched image pairs, LiDAR point cloud projections, camera orientations, and overlap histograms in a top-down view. As part of this thesis, minor refinements were made to this toolkit to improve usability and support the Etna data format.

The final ground truth loop closures were defined as image pairs satisfying the overlap and temporal separation thresholds ($\tau_o = 0.6$, $\tau_t = 100$ ms), ensuring that only true revisits—not consecutive frames—were considered. The temporal constraint τ_t was enforced directly in the script using the timestamp metadata provided in the Etna dataset.

The output of the script for evaluation data consists of a `.csv` and a `.pkl` file, where each row contains a query image, a matched image, and the corresponding overlap score. The `.csv` format facilitates inspection, while the `.pkl` file is used directly by the pipeline during evaluation. These files are later employed to assess image retrieval precision at various top-*k* levels by comparing predicted matches against this ground truth. In total, 794 query images were selected for evaluation.

For training purposes, specifically for fine-tuning DINOv2, a separate script was used to generate triplets in the format `[anchor, positive, negative]`, where each element stores the image path. From each image path, additional metadata—such as the corresponding LiDAR point cloud, pose, and orientation relative to North—can later be retrieved by indexing against the full dataset. This capability facilitates downstream alignment and analysis without requiring reprocessing of raw sensor data.

The selection of positive and negative examples was based on the same viewpoint overlap computation (i.e., `compute_overlap_v1`), but with different thresholding:positives were defined as image pairs with an overlap score above a new threshold $\tau_o^+ = 0.7$, and negatives as those below $\tau_o^- = 0.1$. This separation was designed to ensure that training samples reflected both clear matches and strong mismatches. This procedure yielded 219,460 training triplets, which were split into 175,568 for training (80%) and 43,892 for validation (20%).

While an alternative version of the overlap function (`compute_overlap_v2`) based on polygonal field-of-view intersections—was also implemented and evaluated, it proved overly sensitive to field-of-view assumptions and

consistently failed to capture many visually overlapping pairs. Therefore, `compute_overlap_v1` was ultimately selected for its better alignment with perceptual overlap in the Etna dataset.

4.2 Feature Extraction and Aggregation Pipeline

This section details the core representation learning pipeline used to extract and aggregate visual features from raw images. The process begins with DINOv2, a self-supervised Vision Transformer model, used to extract rich patch-level embeddings from each frame. These features are then either retained in their raw form for geometric tasks or aggregated into compact global descriptors for efficient image retrieval. The section also explains the rationale and setup behind fine-tuning DINOv2 on the Etna dataset, followed by the integration with the SALAD module for descriptor aggregation. Both components are evaluated in terms of their compatibility, training behavior, and final role within the complete localization system.

4.2.1 DINOv2 Feature Extraction and Fine-Tuning

For feature extraction, this project uses the DINOv2 ViT-Base model (`dinov2_vitb14_reg`), loaded from Meta's model repository. The model comprises 12 transformer blocks with an embedding dimension of 768. During training, the final three transformer blocks are unfrozen to enable fine-tuning, while earlier layers remain fixed to preserve general-purpose representations.

Patch embeddings are extracted from the final three layers of the transformer. Specifically, each layer output is sliced to exclude the [CLS] token and retain only the patch-level features. These are then concatenated along the feature dimension and normalized. For downstream applications, two types of embeddings are used:

- **Pose estimation and geometric alignment:** the raw patch embeddings from the last three layers of DINOv2 are used directly without aggregation. These embeddings retain full spatial resolution and are concatenated across layers to form a rich representation of local features.
- **Image retrieval refinement:** the same patch embeddings are averaged across the spatial dimension to produce a compact global descriptor per image, allowing efficient similarity computation.

To qualitatively assess the semantic structure encoded in the DINOv2 patch embeddings, a Principal Component Analysis (PCA) visualization was applied some of the Etna dataset images [21]. The resulting color-coded projection, shown in Figure 4.1, reveals distinct spatial patterns corresponding to scene components. Patches associated with the mountain appear in blue, those covering the sky in green, and those on the sandy terrain in red and orange. Notably, stones scattered across the ground are rendered in colors that differ from the dominant three regions, suggesting that DINOv2 is able to distinguish fine-grained elements within complex environments. This confirms the model's strong capability for capturing semantically meaningful spatial features, a property critical for both loop closure detection and pose estimation in perceptually ambiguous scenes.

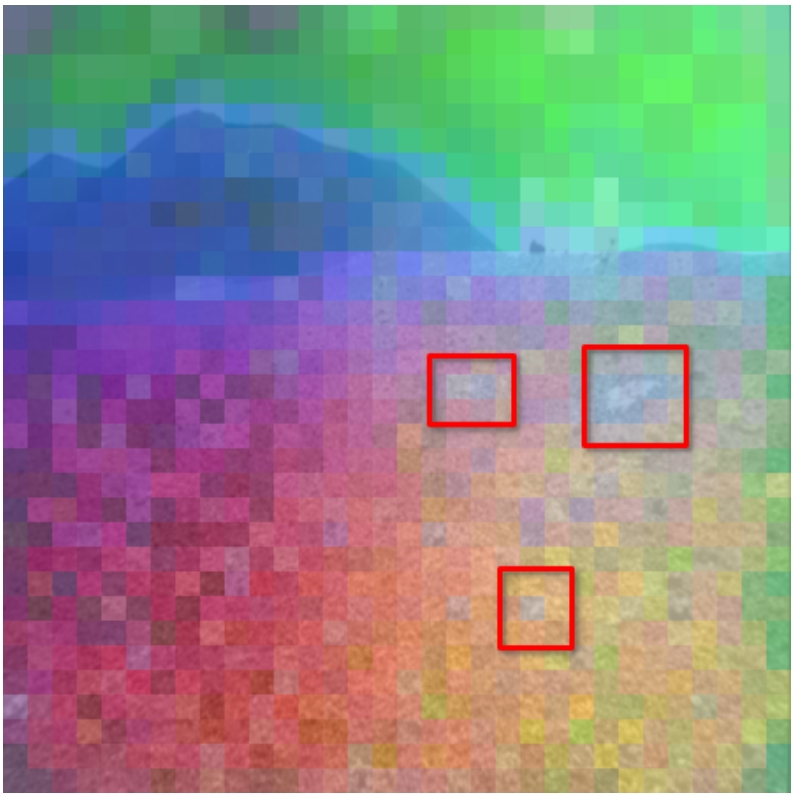


Figure 4.1: PCA visualization of DINOv2 patch embeddings overlaid on an Etna dataset image. Colors correspond to the top three PCA components of patch features from the last transformer layer. The model differentiates semantically meaningful regions: the mountain (blue), sky (green), and sandy floor (red/orange). Importantly, scattered stones on the terrain are rendered in distinct colors, indicating that DINOv2 captures subtle visual elements beyond broad categories.

These DINOv2 embeddings are saved as `.pkl` files containing a dictionary that maps each image path to its corresponding feature tensor and timestamp, enabling fast retrieval during evaluation and inference. The final feature tensor for each image varies by usage: the patch-based embeddings used in pose estimation have a shape of `[n_patches, 2304]`, while the global descriptors used for retrieval have shape `[2304]`.

Fine-tuning of the DINOv2 model was conducted using 219,460 triplets generated from the Etna dataset, split into 175,568 training and 43,892 validation examples. Each triplet includes an anchor, a positive (with overlap > 0.7), and a negative (with overlap < 0.1), identified using the `compute_overlap_v1` function and a minimum temporal separation of

100 ms. During training, data augmentation was applied on-the-fly to improve generalization. This included random variations in brightness, contrast, saturation, and hue, as well as resized cropping. These transformations helped simulate lighting and appearance changes commonly encountered in real-world scenarios.

Training was performed on a dual-GPU Quadro GV100 server (32 GB memory per GPU) using the AdamW optimizer with a learning rate of 1×10^{-5}, batch size of 8, and the triplet margin loss with a margin of 0.2. Early stopping with a patience of 2 epochs was used to prevent overfitting. Training logs were saved to CSV, and a loss curve was plotted to monitor progress. The best-performing checkpoint (based on validation loss) was retained and used in the final system.

The training and validation loss curves in Figure 4.2 show a consistent decrease in training loss over epochs, indicating that the model is effectively learning from the training data. The validation loss initially decreases, mirroring the training loss, but starts to fluctuate slightly after epoch 3, suggesting minor overfitting. However, this fluctuation is expected due to the use of early stopping with a patience of 2, which allows training to continue for a few epochs after the validation loss stops improving, helping to avoid premature stopping. Overall, the trend of the validation loss remains relatively flat and low, staying close to the training loss, which indicates that the model is generalizing well to the validation data and not significantly overfitting.

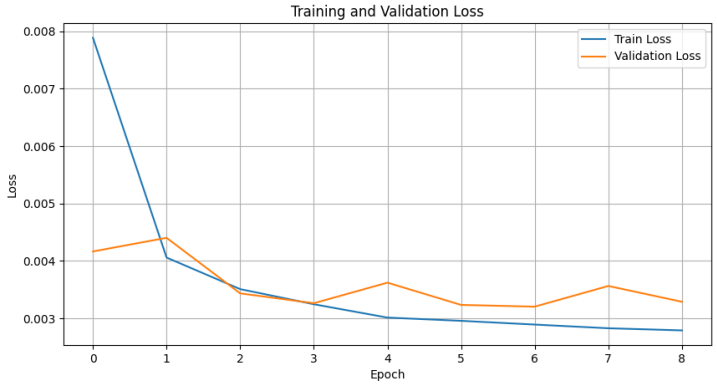


Figure 4.2: Training and validation loss curves during fine-tuning of DINOv2. Early stopping is triggered after the 9th epoch when validation loss no longer improves.

The fine-tuned DINOv2 model was ultimately adopted in the final pipeline, both as a feature extractor for patch embeddings and as input to the aggregation

module. However, although additional experiments were conducted to adapt SALAD to aggregate concatenated embeddings from the last three transformer layers, these modifications led to worse performance compared to the original SALAD model. As a result, the original SALAD architecture, using single-layer features with the [CLS] token, was retained in the final design, while the fine-tuned DINOv2 weights were still used to initialize the base encoder.

4.2.2 Connecting DINOv2 to SALAD

To integrate DINOv2 as a feature extractor within the image retrieval pipeline, the model's output had to be aligned with the input expectations of the SALAD aggregation module. The original SALAD architecture was designed to operate on patch embeddings from Vision Transformers, typically from a single layer, including the [CLS] token.

Initially, the standard version of SALAD was used without modification, applying it to the output of the fine-tuned DINOv2 model. In this configuration, patch embeddings (including the [CLS] token) from the final transformer layer were passed directly into the aggregation module. This setup produced reliable results and served as the baseline.

To explore whether richer features could improve performance, a custom version of SALAD was implemented that aggregates concatenated patch embeddings from the last three transformer layers of DINOv2. This involved modifying the forward pass of the DINO model to output a 3D tensor of shape `[B, 2304, H/14, W/14]`, constructed by stacking patch tokens from the last three layers (each with dimension 768). Correspondingly, the aggregation head of SALAD was adapted to accept an input with 2304 channels (i.e., `3` layers $\times$ `768` dim) and was configured with the following hyperparameters:

- `num_channels = 2304`
- `num_clusters = 64`
- `cluster_dim = 128`

Despite these adjustments, the modified aggregation approach did not yield improved performance. It was observed that including multi-layer features introduced more noise than discriminative power, possibly due to redundancy or misalignment across layers. Consequently, the original SALAD model was retained for the final pipeline. However, it was used

in conjunction with the fine-tuned DINOv2 backbone, ensuring that the aggregation benefited from the improved feature representations learned during training.

4.2.3 Re-training SALAD

To explore whether adapting the SALAD aggregation module to better align with DINOv2 feature representations could yield performance improvements, a modified version of SALAD was trained using patch embeddings concatenated from the last three transformer layers of the fine-tuned DINOv2 model. This adjustment increased the input dimensionality from 768 to 2304, requiring structural changes to the aggregation head.

The training followed the protocol described in the original SALAD paper and used the same datasets: Mapillary Street-Level Sequences (MSLS), GSV Cities, and Pittsburgh250k. These datasets offer diverse, large-scale urban environments with GPS-based ground truth, making them suitable for training global image descriptors using triplet loss.

Training was conducted on the same hardware as used for DINOv2 fine-tuning—a dual-GPU Quadro GV100 workstation with 128GB RAM. The process was stable and completed successfully.

However, when tested on the Etna dataset, the re-trained SALAD module did not outperform the original version. In fact, the modified model showed slightly reduced retrieval and pose estimation performance, suggesting that the higher-dimensional feature input may have introduced noise or led to overfitting on irrelevant patterns in the training data.

As a result, the final pipeline retained the original SALAD architecture, which aggregates features from the last transformer layer using the `[CLS]` token. This version, combined with the fine-tuned DINOv2 backbone, provided the most consistent and robust performance in the downstream tasks.

4.3 Descriptor Storage and Retrieval Infrastructure

To support efficient large-scale image retrieval, all feature descriptors used in this system are precomputed and stored as `.pkl` files. These files contain dictionaries mapping each image path to its corresponding descriptor vector and associated timestamp. For organizational clarity and modularity, descriptors are saved in separate folders depending on their role in the pipeline.

The retrieval process is implemented using Facebook AI Similarity Search (FAISS) [27], specifically the `IndexFlatIP` index with L2-normalized descriptors. This configuration enables fast and scalable approximate nearest neighbor search based on cosine similarity, which aligns well with the normalized output of the DINOv2 and SALAD embedding models.

A two-stage retrieval strategy is employed to improve robustness:

- **Coarse Retrieval:** Global descriptors aggregated using the SALAD module are used to perform an initial ranking of the top-k candidate matches. This step captures high-level visual similarity and efficiently narrows the search space.
- **Fine Re-ranking:** The top candidates from the coarse stage are then re-ranked using a finer-grained descriptor. Specifically, DINOv2 patch embeddings from the last three layers are averaged to produce an independent global descriptor. These refined descriptors help disambiguate visually similar scenes and improve match quality.

This two-stage pipeline balances speed and accuracy: the SALAD-based global descriptor provides efficient broad filtering, while the DINOv2-based refinement stage enhances precision, particularly in environments with high perceptual aliasing.

4.4 Pose Estimation Pipeline

This section details the multi-modal approach developed to estimate the relative 6-DoF pose between two frames in the Etna dataset. The pipeline integrates high-dimensional visual descriptors extracted from a fine-tuned DINOv2 model with semantic 3D features from the Sonata model. These complementary features are aligned and fused to enable robust point correspondence matching, which is then used for pose estimation through geometric optimization techniques.

4.4.1 Visual Embedding Projection to 3D

The process begins by extracting patch-level embeddings from the final three layers of the DINOv2-base Vision Transformer (ViT-B/14), which has been fine-tuned on Etna data using triplet loss. These embeddings are normalized and concatenated across layers, yielding a $[n\_patches, 2304]$ tensor per image.

For pose estimation, the goal is to associate these descriptors with real-world 3D coordinates.

To achieve this, the corresponding LiDAR point cloud for each image is projected into the image plane using the camera's intrinsic parameters. Each valid 3D point is mapped to a 2D pixel coordinate and then associated with the closest DINO patch based on its spatial location in the image grid (560×560 resolution, 14×14 patch structure). This enables each 3D point to inherit the high-level semantic descriptor from the corresponding visual patch.

This method provides a dense 3D embedding set per frame, where each point carries both geometric location and a visual descriptor derived from the transformer. Figure 4.1, demonstrates the semantic richness of these features using a PCA projection of patch embeddings over an Etna image. Blue-toned patches correspond to mountainous regions, green to sky, and red/orange to sandy terrain. Interestingly, small rocks and discrete elements on the ground appear in colors distinct from these major classes, illustrating DINO's fine-grained discrimination capabilities.

4.4.2 Sonata 3D Feature Extraction

To complement the visual embeddings, each LiDAR point cloud is also processed through the Sonata model to produce dense 3D descriptors [73]. These embeddings encode both semantic and geometric properties of the environment, providing a powerful modality for matching when visual features may fail (e.g., due to illumination or viewpoint changes).

Figure 4.3 shows a PCA projection of Sonata features. Semantic patterns naturally emerge, rocky formations are clearly separated from flatter ground regions. The blacked-out points correspond to filtered areas where the first principal component is negative, which are excluded following the visualization strategy in the original DINOv2 work [34].

4.4.3 Feature Fusion and Correspondence Matching

After extracting descriptors from both modalities, visual and geometric features are fused at the point level. Each 3D point descriptor is formed by concatenating its DINO-derived embedding and its Sonata descriptor, resulting in a hybrid feature that captures both appearance and geometry.

To find correspondences between a query and candidate frame, cosine similarity is computed between each pair of fused descriptors. Matches are selected using the Hungarian algorithm to enforce one-to-one correspondence.

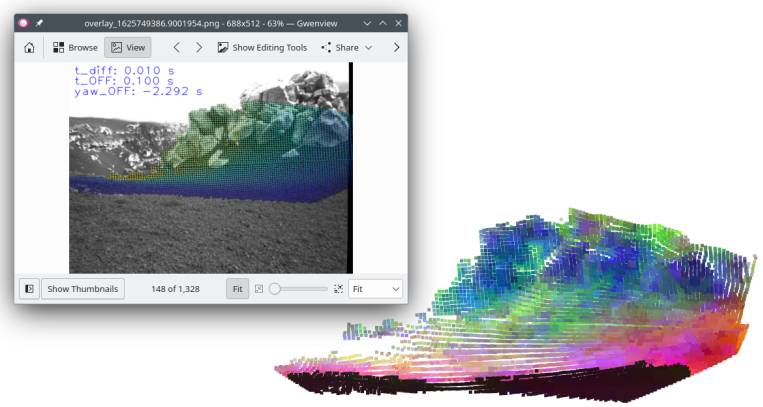


Figure 4.3: PCA projection of Sonata LiDAR descriptors. Distinct regions in the environment such as rock formations (blue) and ground surfaces (green/pink) are semantically separated. Points with ambiguous or low-information content are excluded (black).

A threshold (e.g., 0.9) is applied to filter out weak or ambiguous matches.

Figures 4.4 and 4.5 highlight the limitations of using only a single modality: DINOv2 descriptors tend to overmatch, resulting in dense but noisy correspondences; Sonata descriptors, while geometrically robust, may lack discrimination in visually complex areas. Figure 4.6 shows the benefit of fusion—fewer matches are produced, but they are significantly more accurate and geometrically coherent.

4.4.4 Pose Estimation and Re-Ranking

Once a set of high-confidence correspondences has been established, the relative pose between the query and candidate frame is estimated. The initial transformation is computed using a RANSAC-based solver that fits a rigid transformation to the 3D matched points. This helps reject outliers and produce a stable estimate.

Following RANSAC, an optional ICP refinement step is applied. This further aligns the two point clouds by minimizing geometric error. The ICP fitness score, defined as the proportion of inlier points after alignment, is computed and used to filter out geometrically invalid matches.

After pose estimation, the candidates are re-ranked based on their spatial proximity to the query, specifically, the angular difference in their yaw orientations. This prioritizes geometrically consistent results while preserving

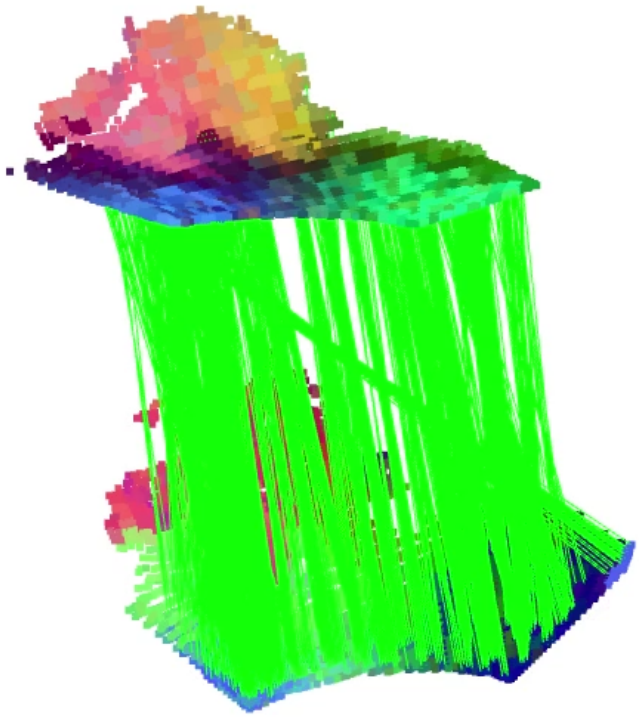


Figure 4.4: Point correspondences using only DINOv2 visual features. Many incorrect matches are observed due to lack of geometric constraints.

robustness against false positives.

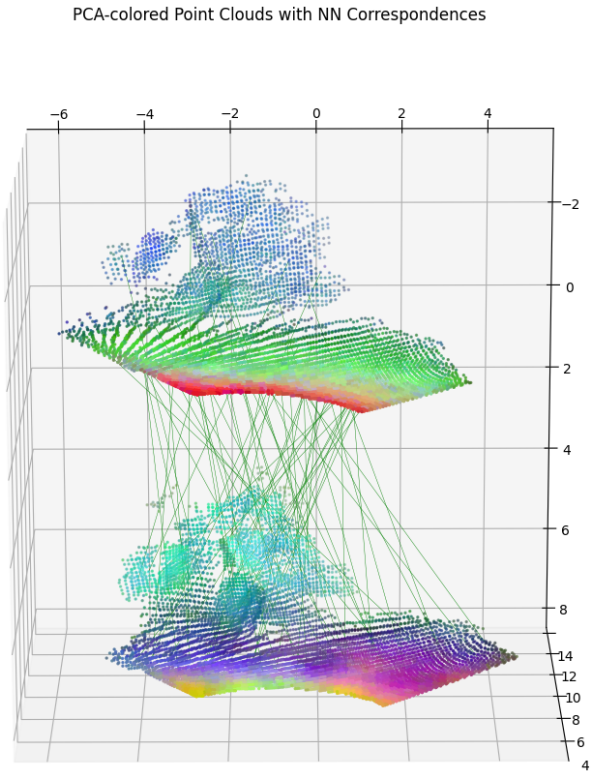


Figure 4.5: Point correspondences using only Sonata 3D features. Mismatches occur in low-texture or structurally repetitive areas.

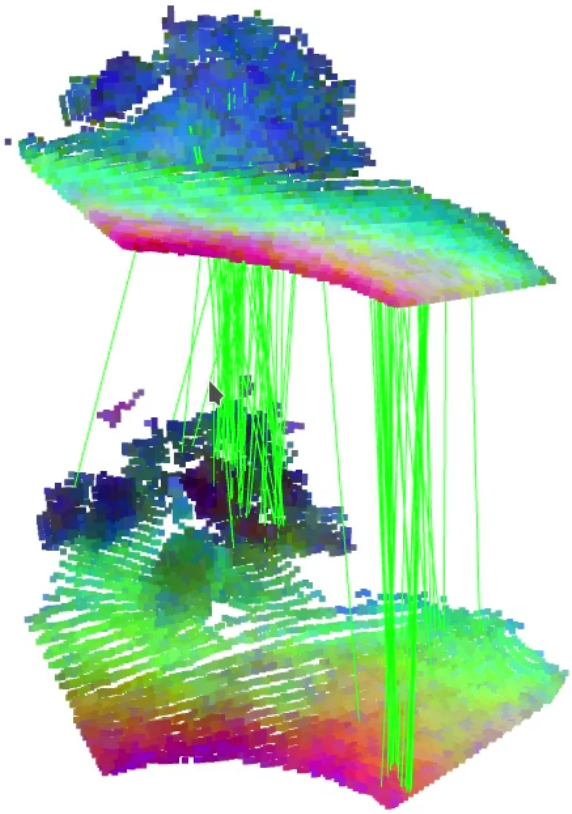


Figure 4.6: Point correspondences using fused DINOv2 and Sonata features. Matches are more accurate, consistent, and less noisy than those using either modality alone.

Chapter 5

Results and Analysis

This chapter presents a comprehensive evaluation of the proposed multi-modal pipeline for image-based place recognition and relative pose estimation. Section 5.1 introduces the major quantitative results across both core tasks: image retrieval and pose estimation. Section 5.1.1 analyzes the retrieval performance in terms of precision at top-k ranks and runtime across various baselines, highlighting the benefits of fine-tuning and aggregation. Section 5.1.2 evaluates pose estimation accuracy using yaw and translation errors, comparing the proposed fusion model against transformer-based and regression-based baselines. In Section 5.2, the model's consistency is analyzed through pose coverage, error distributions, and qualitative examples. Finally, Section 5.3 addresses the validity of the evaluation metrics and experimental setup, confirming their alignment with the system's intended deployment scenarios in unstructured environments.

5.1 Major Results

This section presents the main quantitative results of the proposed visual place recognition and pose estimation pipeline. The evaluation focuses on two key tasks: image retrieval and pose estimation. In both cases, the performance of the proposed method is compared against a variety of existing baselines, including traditional handcrafted descriptors, recent transformer-based methods, and recent neural models tailored to place recognition and localization.

5.1.1 Image Retrieval Performance

The image retrieval task was evaluated by measuring precision at top-k ranks, specifically at $k = 1$, $k = 5$, and $k = 10$. Precision at top-k is defined as the proportion of retrieved images (among the top k) that correspond to ground truth matches. A true match is defined according to the ground truth thresholds ($\tau_o = 0.6$) as an image pair with an overlap score above τ_o, computed using the `compute_overlap_v1` function introduced in Section 4.1. All descriptors used in this task are precomputed and stored in the format described in Chapter 4, and retrieval is performed via cosine similarity search using a FAISS index with L2-normalized vectors.

Table 5.1 summarizes the retrieval performance across several models, including CNN-based methods such as NetVLAD [9], transformer-based baselines like TransVPR [36], and different configurations of DINOv2-based descriptors [34], both with and without the SALAD aggregation module [26]. Runtime measurements (in milliseconds) are also included to assess computational efficiency.

graphicx

Table 5.1: Image Retrieval Results: Precision at Top-k and Average Retrieval Time

Model	Precision@1	Precision@5	Precision@10	Time (ms)
NetVLAD	0.4395	0.4237	0.4033	1249.89
TransVPR	0.4534	0.4325	0.4126	392.39
DINOv2 (s) (last 3 layers)	0.2179	0.2033	0.1897	370.87
DINOv2 (b) (CLS Token)	0.5982	0.5471	0.5161	1123.11
DINOv2 (b) (last 3 layers)	0.6474	0.6081	0.5838	1216.88
SALAD (pretrained)	0.6378	0.7048	0.6792	369.82
Proposed Model (pretrained SALAD + pretrained DINOv2)	0.7116	0.6897	0.6795	389.71
Proposed Model (pretrained SALAD + fine-tuned DINOv2)	**0.7569**	**0.7332**	**0.7090**	476.57
Proposed Model (retrained SALAD + fine-tuned DINOv2)	0.7141	0.6965	0.6712	578.21

The best performing configuration is the proposed method combining original SALAD with the fine-tuned DINOv2 backbone. It achieved a precision of 75.69% at top-1, 73.32% at top-5, and 70.90% at top-10. This significantly outperforms CNN-based baselines such as NetVLAD and even transformer-based methods like TransVPR, while offering competitive inference time. Fine-tuning DINOv2 provided a clear benefit over using the base encoder out-of-the-box. However, attempts to improve performance by retraining SALAD with modified input dimensions resulted in a drop in precision, indicating that the original architecture is better suited for this task when paired with a strong image encoder.

To better visualize the trade-off between retrieval accuracy and computational efficiency, Figure 5.1 presents a scatter plot of Precision@1 versus average retrieval time for all evaluated methods. Each point represents a different model configuration. The results show that the proposed method combining pretrained SALAD with fine-tuned DINOv2 (SALAD+DINO-ft) offers the best balance, achieving the highest precision while maintaining a reasonable retrieval time. In contrast, methods like NetVLAD and even transformer-based baselines such as TransVPR exhibit lower accuracy despite similar or higher computational costs. Interestingly, models that solely rely on DINOv2 without aggregation (e.g., DINOv2-b-CLS and DINOv2-b-3L) perform well in terms of precision but suffer from significantly higher inference times, making them less suitable for time-sensitive applications. Overall, the plot highlights the efficiency and effectiveness of combining compact aggregation (SALAD) with strong pretrained or fine-tuned visual descriptors.

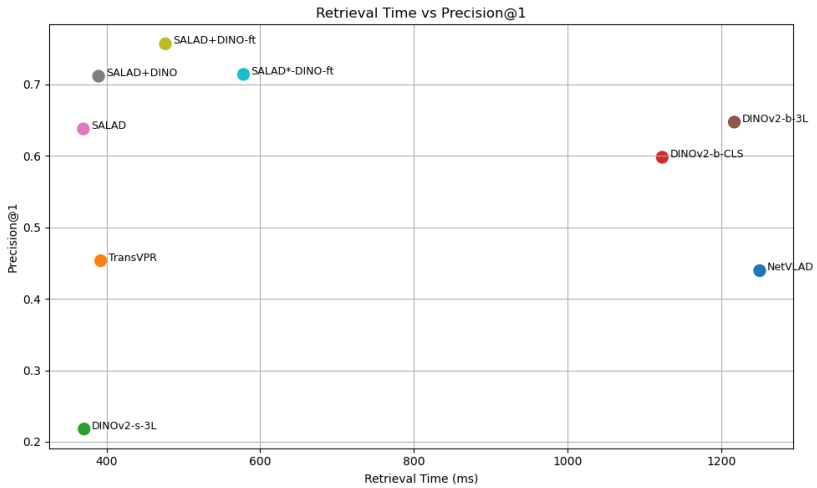
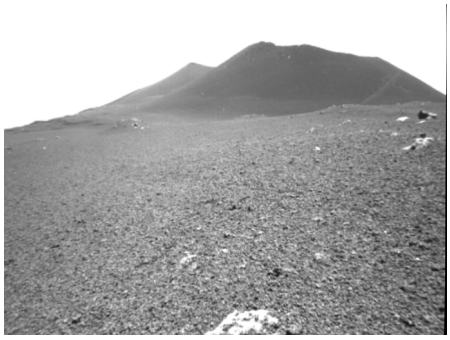


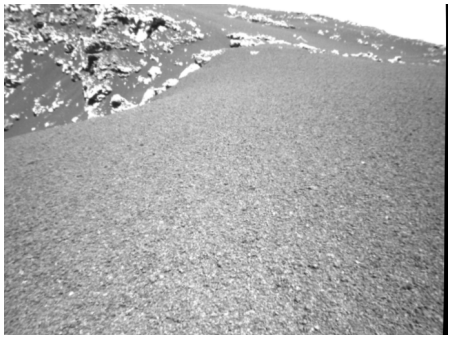
Figure 5.1: Trade-off between retrieval time (in milliseconds) and Precision@1 for all evaluated methods. Top-left positions represent models that are both accurate and efficient. The proposed method (SALAD+DINO-ft) achieves the best balance, outperforming traditional baselines.

To complement the quantitative results presented above, Figure 5.2 shows three examples of image retrieval outcomes using the proposed method (SALAD + fine-tuned DINOv2). These examples highlight the ability of the model to recognize scene similarity under changes in viewpoint and illumination, as well as one failure case where perceptual similarity leads to an incorrect match.

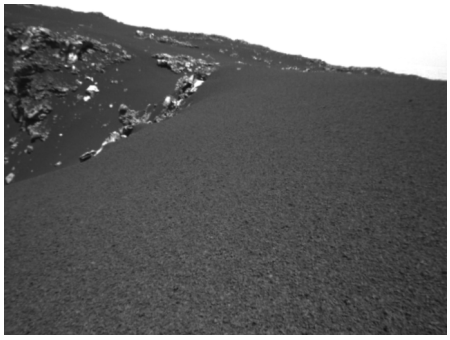
(a) Query image (viewpoint shift)



(b) Retrieved image: correct match despite partial mountain occlusion



(c) Query image (lighting variation)



(d) Retrieved image: correct match under illumination change, crater visible

(e) Query image (perceptually ambiguous)

(f) Retrieved image: incorrect match due to similar horizon and rock patterns

Figure 5.2: Examples of image retrieval outcomes using the proposed model. The first row shows a successful match under a viewpoint change. The second row shows a match under both viewpoint and lighting variations. The third row shows a failure case, where similar horizon and rock features caused a false positive retrieval, despite not satisfying the overlap threshold.

5.1.2 Pose Estimation Performance

To evaluate the pose estimation performance of the proposed system, we report the average yaw error (in degrees), average translation errors in the X and Y directions (in meters), runtime per query (in milliseconds), and the number of poses successfully estimated. A successful pose is defined as one that passes RANSAC verification with sufficient feature correspondences.

Table 5.2 presents the results for several baselines, including traditional 3D feature descriptors (FPFH), 2D and 3D transformer-based models (LoFTR, SONATA), and the latest regression-based model (reloc3r). The proposed model, which fuses projected DINOv2 and SONATA features, achieves high pose coverage and good accuracy, but the best overall performance in terms of both precision and computational efficiency is achieved by reloc3r.

Table 5.2: Pose Estimation Results: Average Errors, Total Poses Estimated, and Inference Time

Model	Yaw Error (°)	DX Error (m)	DY Error (m)	Poses Estimated	Time (ms)
FPFH + RANSAC	46.82	8.23	14.27	**1560**	12233.82
DINO-LiDAR + RANSAC	25.10	8.40	14.27	**1560**	5686.82
LoFTR + RANSAC	11.40	**4.13**	**6.92**	744	249.36
DINO-Patches + RANSAC	17.13	8.06	14.05	1353	948.30
SONATA + RANSAC	16.36	8.86	15.30	1066	3572.05
Reloc3r	**8.15**	8.31	14.19	**1560**	**133.76**
Proposed Model (DINO + SONATA)	8.20	8.44	14.24	**1560**	3114.33

While the proposed model achieves strong accuracy and high pose coverage across all queries, it is not the best performer in terms of runtime. The reloc3r model not only achieves the lowest yaw error but also matches the proposed method in the number of successful poses while running an order of magnitude faster. Nevertheless, the proposed system offers an interpretable, matching-based alternative that does not rely on end-to-end supervision and maintains competitive accuracy under a fusion-based architecture.

To provide a finer-grained view of model reliability, we report the percentage of estimated poses whose yaw and translation errors fall within a set of predefined thresholds. This allows us to analyze how often models predict poses that are geometrically close to the ground truth.

Table 5.3 presents yaw angle error percentages for thresholds at 2°, 3°, 5°, and 10°. Table 5.4 complements this by showing the proportion of translation errors in both the X and Y directions falling under 1, 2, 3, 5, and 10 meters.

This breakdown reveals that while LoFTR exhibits the best translational accuracy, reloc3r performs best overall in angular precision and robustness across all thresholds. The proposed model shows competitive performance,

especially in yaw estimation, and maintains close performance to reloc3r in terms of translation reliability.

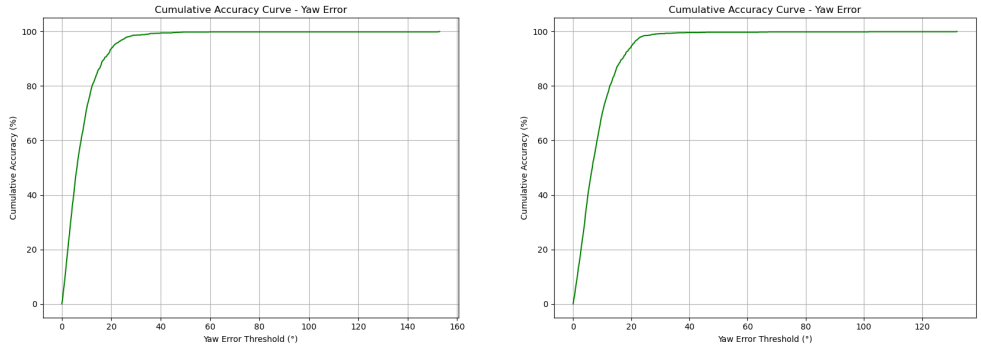
Table 5.3: Percentage of Estimated Poses with Yaw Error Below Thresholds

Model	$< 2°$	$< 3°$	$< 5°$	$< 10°$
LoFTR + RANSAC	13.31	20.03	32.66	60.48
Reloc3r (2025)	16.22	25.00	41.60	71.22
Proposed model	14.29	22.56	39.10	69.94

Table 5.4: Percentage of Estimated Poses with Translation Error in X and Y Below Thresholds

Model	**DX Error <**					**DY Error <**				
	1m	2m	3m	5m	10m	1m	2m	3m	5m	10m
LoFTR + RANSAC	22.58	36.83	51.48	72.85	91.67	19.35	34.68	43.28	56.99	79.03
Reloc3r (2025)	9.36	18.14	26.47	44.49	66.92	11.09	20.60	28.33	37.63	57.05
Proposed model	8.65	18.21	27.37	42.44	65.45	11.09	20.38	27.88	36.99	57.12

To further investigate the angular estimation performance of the proposed model, we analyze cumulative accuracy curves based on yaw error thresholds. Figure 5.3 compares the cumulative accuracy between the proposed method and the Reloc3r baseline. The proposed model consistently yields higher accuracy at lower yaw error thresholds, with over 90% of poses estimated within 15° of ground truth, slightly outperforming Reloc3r in the sub-10° region. While both models converge near 100% accuracy at larger thresholds, the proposed system reaches saturation earlier, indicating greater reliability in producing precise yaw predictions. This improved precision at tighter thresholds is particularly important for downstream tasks such as visual localization and path planning, where even small orientation errors can lead to significant downstream effects.



(a) Reloc3r Baseline (b) Proposed Model

Figure 5.3: Cumulative accuracy curves of yaw estimation error for Reloc3r and the proposed method. Higher values at lower thresholds indicate better angular precision. The proposed model (b) reaches saturation slightly more quickly and maintains stronger accuracy in the critical sub-10° error region.

Additionally, a box plot of yaw errors for the three top-performing models—My Model, Reloc3r, and LoFTR—is shown in Figure 5.4. The distribution reveals that while all three models have a similar interquartile range (IQR), Reloc3r displays the lowest median yaw error, confirming its overall advantage in angular precision. The proposed model shows a slightly higher median but with fewer extreme outliers than LoFTR and reloc3r. LoFTR, despite having a competitive IQR, exhibits the widest spread of high-error outliers, indicating occasional large deviations in orientation estimation. These findings reinforce earlier metrics and highlight the consistency of Reloc3r and the robustness of the proposed method, particularly in avoiding catastrophic angular failures.

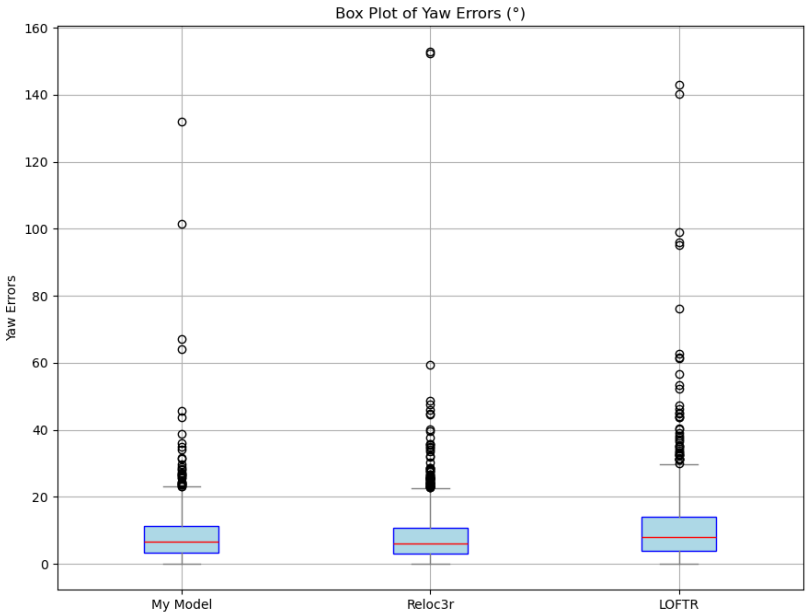


Figure 5.4: Box plot of yaw errors (in degrees) for the top three models. The median error is lowest for *reloc3r*, while the proposed model shows fewer extreme outliers than *LoFTR*.

5.2 Reliability Analysis

Reliability refers to the consistency and stability of the system's performance across different inputs and evaluation metrics. The results presented in Section 5.1 demonstrate that the proposed model performs robustly across a range of image retrieval and pose estimation scenarios.

For image retrieval, consistent improvements in precision across top-k ranks (Table 5.1) indicate that the model maintains strong matching performance regardless of the retrieval k. The method performs reliably across both challenging viewpoint and illumination variations, as evidenced by qualitative examples (Figure 5.2).

In pose estimation, the proposed model achieves similar coverage as reloc3r, successfully estimating 1,560 poses, while maintaining acceptable yaw and translation errors (Table 5.2). The box plot in Figure 5.4 shows that the proposed method has a stable yaw error distribution with limited extreme outliers, suggesting robustness across different queries. Furthermore, the cumulative accuracy curves (Figure 5.3) show consistent pose accuracy across increasing error thresholds, reinforcing the model's reliability in angular

estimation.

Overall, the evaluation metrics demonstrate that the system delivers reproducible results under diverse and realistic conditions, confirming its reliability.

5.3 Validity Analysis

Validity concerns whether the evaluation setup and metrics truly measure what the system is intended to achieve. In this work, both tasks—image retrieval and pose estimation—are evaluated using metrics that align closely with their intended real-world functions.

For image retrieval, a retrieved image is considered a correct match only if it exceeds an overlap threshold of 0.6 with the query image, as defined in Section 4.1. This threshold ensures that the system retrieves not just visually similar scenes, but ones that are truly spatially relevant, which enhances the construct validity of the evaluation. The inclusion of failure cases and correct matches under viewpoint and lighting changes (Figure 5.2) further supports the method's validity in real-world conditions.

Pose estimation validity is ensured by evaluating against ground truth poses using standard geometric metrics: yaw angle and translation error in meters. The use of multiple error thresholds (Tables 5.3 and 5.4) provides a nuanced view of the system's precision. Additionally, RANSAC-based inlier filtering ensures that only geometrically consistent poses are considered, reinforcing the correctness of each estimated transformation.

By combining semantic similarity for image retrieval with geometric correctness for pose estimation, the system's evaluation pipeline accurately reflects the core objectives of robust place recognition and localization.

Chapter 6

Discussion

This chapter reflects on the findings presented in Chapter 5, offering a broader interpretation of their implications for visual localization in unstructured and GNSS-denied environments. Section 6.1 evaluates the practical relevance of the proposed system for real-world field robotics, highlighting its strengths in environments where traditional methods fail. Section 6.2 compares different localization paradigms—regression, matching, and handcrafted approaches—emphasizing the trade-offs between accuracy, interpretability, and efficiency. Section 6.4 outlines unexpected behaviors and counterintuitive outcomes observed during experimentation, offering insights into model fragility and generalization. Section 6.5 positions the contributions of this thesis within the broader research landscape, identifying the niche it fills and its applicability in future autonomous systems. Together, these sections provide a deeper understanding of the system's capabilities, limitations, and potential impact.

6.1 Applicability in Unstructured and GNSS-Denied Environments

The experimental results presented in this thesis show that the proposed model is well-suited for visual place recognition and pose estimation in challenging, low-structure environments. Unlike many existing works that are evaluated in urban or indoor settings—such as cityscapes, university campuses, or office corridors—this project targets a more demanding scenario: a natural, low-texture terrain that closely resembles planetary or remote field environments.

This distinction is significant because urban datasets typically contain semantic regularities like roads, buildings, and signs, which offer strong

visual cues for models trained in those domains. Consequently, descriptor-only baselines like NetVLAD [9] and TransVPR [36], which perform well on standard benchmarks, show poor generalization when applied to the Etna dataset. Their reliance on high-level semantics becomes a limitation in environments where such cues are absent or ambiguous.

In contrast, the proposed image retrieval system—based on a fine-tuned DINOv2 encoder and the SALAD aggregation module—achieved a top-1 retrieval precision of 75.69% (Table 5.1). This performance confirms that self-supervised features, combined with efficient and adaptive aggregation, can significantly improve robustness in visually sparse conditions. Importantly, the retrieval system also maintains reasonable computational efficiency, with an average inference time under 500 ms, which is compatible with real-time onboard use in many robotic platforms (Figure 5.1).

For tasks like SLAM, reliable image retrieval is essential for loop closure, which helps correct drift and ensures global map consistency. Failure in recognizing a revisited place may result in serious localization errors, particularly when GNSS signals are unavailable. Hence, accurate and efficient retrieval is not just a performance goal but a prerequisite for long-term autonomous navigation.

The same applies to pose estimation. While regression-based models like Reloc3r [63] showed the best accuracy and runtime (Table 5.2), the proposed matching-based system using projected DINOv2 features and SONATA achieved high pose coverage and competitive yaw accuracy, with fewer catastrophic outliers than LoFTR (Figure 5.4). This is particularly important in cratered or texture-sparse landscapes, where handcrafted descriptors like FPFH [77] often fail to yield enough reliable matches for geometric solvers.

In this setting, matching-based techniques provide more interpretable and failure-resilient solutions, as they build pose estimates directly from observable similarities. This is especially relevant in safety-critical scenarios such as planetary exploration, where explainability and modularity are as important as raw performance. While the proposed system is slower than regressive alternatives, it offers transparency, robustness, and adaptability—attributes that are essential for autonomous systems operating in unknown, unstructured environments.

To summarize, the findings of this work reinforce the need to design localization methods that are not only accurate but also interpretable and reliable in visually ambiguous or GNSS-denied settings. This is essential for real-world field robotics applications, including planetary exploration, search and rescue, and long-range autonomous navigation in remote areas.

6.2 Comparative Analysis of Localization Methodologies

This section presents a structured comparison between the different categories of localization systems evaluated in this thesis, focusing on both image retrieval and pose estimation. The goal is to analyze their respective strengths, limitations, and practical trade-offs in terms of interpretability, generalization, runtime, and robustness.

6.2.1 Image Retrieval Systems.

The image retrieval methods evaluated in this thesis differ primarily in how they aggregate features into compact descriptors. Aggregation strategy plays a crucial role in retrieval robustness and generalization, particularly under the visually sparse and ambiguous conditions targeted in this work.

VLAD-based aggregation methods include both traditional approaches like NetVLAD [9] and more advanced variants like SALAD [26]. NetVLAD uses soft assignment of CNN-based local features to cluster centroids and computes residuals, offering compact descriptors but limited adaptability to novel domains. In contrast, SALAD integrates DINOv2 [34] features with a learned VLAD-style framework that replaces k-means clustering with a differentiable assignment via the Sinkhorn algorithm and introduces a "dustbin" cluster to ignore uninformative features. It also includes fully connected layers for dimension reduction and a global token fusion step. These enhancements improve robustness to viewpoint and appearance variations, making SALAD particularly effective in unstructured environments.

Attention-based aggregation is used in TransVPR [36], which applies a transformer architecture to aggregate image patch features using global self-attention. While effective in semantically rich urban scenes, its reliance on structured content reduces its effectiveness in the natural terrain of the Etna dataset, where semantic cues are sparse.

Global descriptor variants of DINOv2 were also evaluated to assess the standalone capability of self-supervised features. The CLS-token version of DINOv2 base (ViT-B/14) achieved a top-1 precision of 59.82%, while a variant using the average of the last three layers across all patches reached 64.74%, both outperforming conventional baselines. However, the best performance was obtained by combining DINOv2 with SALAD, which reached 75.69% precision. These results, summarized in Table 6.1, highlight the importance of aggregation strategy: combining domain-robust

features with learnable, VLAD-style pooling significantly improves retrieval performance in low-texture settings.

The proposed image retrieval system—based on DINOv2 features aggregated with SALAD—achieved the best top-1 precision (75.69%) on the Etna dataset, significantly outperforming both NetVLAD and TransVPR. This demonstrates the advantage of pairing robust, self-supervised features with learned VLAD-style aggregation tailored for retrieval.

In summary, VLAD-style aggregation remains competitive, especially when enhanced through learning-based methods like SALAD. Attention-based approaches offer semantic richness but struggle in unstructured scenes. The findings highlight the importance of using robust descriptors and adaptive aggregation for generalization to unstructured natural environments.

Table 6.1: Comparison of image retrieval methods evaluated on the Etna dataset. DINOv2 variants use the "base" model (ViT-B/14) with either the CLS token or the average of the last three layers across all patches.

Method	Backbone	Aggregation	Runtime (ms)	Robustness
NetVLAD [9]	VGG-16	Fixed VLAD	Slow	Low
TransVPR [36]	Transformer	Attention Pooling	Fast	Low
DINOv2 (b) CLS Token	ViT-B/14	CLS Token	Slow	Medium
DINOv2 (b) Last 3 Layers	ViT-B/14	Patch Avg (3 Layers)	Slow	Medium–High
DINOv2 + SALAD [26]	ViT-B/14	Learnable VLAD (Sinkhorn)	Fast	High

6.2.2 Pose Estimation Systems.

The pose estimation approaches in this thesis can be categorized based on their core design principles and data dependencies.

Regression-based methods, such as Reloc3r [63], directly learn to predict camera pose from image input via deep networks. These models achieve high pose accuracy and runtime efficiency, as demonstrated by Reloc3r's strong performance on yaw and translation metrics. However, they lack interpretability and are more difficult to diagnose in failure cases, making them less suitable for safety-critical applications where transparency is important.

Matching-based pipelines, including LoFTR [8], DINO-Patches [60], and the proposed DINO+SONATA model, estimate pose by establishing explicit correspondences between query and reference features. These methods are generally more interpretable, as their outputs are based on observable feature matches. The proposed approach, which integrates 2D visual features from DINOv2 with 3D geometric descriptors from SONATA, achieved full coverage across all queries and strong yaw accuracy with fewer

catastrophic outliers than LoFTR.

Handcrafted descriptor methods, such as FPFH [77] combined with RANSAC [16], rely on predefined 3D geometric features and traditional matching. While conceptually simple, they performed poorly in low-texture scenes, frequently failing to produce sufficient inlier matches for pose estimation. These results underline the limitations of purely handcrafted pipelines in complex or unstructured environments.

In summary, regression-based methods offer speed and accuracy but sacrifice transparency. Matching-based systems, particularly those fusing 2D and 3D cues like the proposed DINO+SONATA, strike a balance between robustness and interpretability. Handcrafted pipelines, while efficient in structured scenes, struggle under the environmental conditions explored in this work. A comparative summary of these methods is presented in Table 6.2.

Table 6.2: Comparison of Pose Estimation Methods

Method	Category	Yaw MAE (°)	Runtime (ms)	Interpretability
FPFH + RANSAC [77, 16]	Handcrafted	High	Slow	High
Reloc3r [63]	Regression-based	**Lowest**	**Fastest**	Low
LoFTR [8]	Matching-based	Medium	Medium	Medium
DINO-Patches [60]	Matching-based	Medium	Slow	Medium
Projected ViT (DINO-LiDAR) [79]	Matching-based	High	Slow	High
Projected ViT (DINO+SONATA)	Matching-based	Low	Slow	High

6.3 Insights from Ablations and Variants

This section summarizes key lessons from the ablation studies and model variants evaluated during the thesis. These insights help explain the performance trends observed in Chapter 5 and inform the design decisions made for the final pipeline.

a) **Impact of Fine-Tuning DINO.**
Fine-tuning the DINOv2 backbone on the Etna dataset produced a significant gain in retrieval accuracy. While the proposed model using pretrained DINOv2 and pretrained SALAD reached a top-1 precision of 71.16%, fine-tuning DINOv2 improved this to 75.69%. This highlights the importance of domain adaptation, even for strong self-supervised models. Although the pretrained features were already effective, this performance gap suggests that general-purpose features lack the necessary specificity to handle ambiguous or repetitive terrain, which is common in natural and planetary-like environments.

b) Design Choice: DINO Base Model and Layer Strategy.
The performance of different DINO configurations was also evaluated. DINOv2 (b), which uses a larger ViT backbone, significantly outperformed its small counterpart DINOv2 (s), achieving up to 64.74% precision when using the last three layers compared to only 21.79% for DINOv2 (s). Additionally, using the last three layers and averaging patch-level features provided better results than the standard CLS token (64.74% vs 59.82%). These findings justify the use of DINOv2 (b) with multi-layer aggregation in the final pipeline.

c) Fragility of SALAD when Re-trained.
Attempts to retrain SALAD using the same dataset—after changing the DINO feature aggregation strategy—showed that the model is sensitive to both initialization and training setup. While the original SALAD performed best when used off-the-shelf with DINOv2 (b), retraining led to reduced performance. This suggests that SALAD's optimal performance is closely tied to its original pretraining conditions, and modifying the input representation without adjusting the architecture or carefully tuning hyperparameters can destabilize feature-cluster assignments.

d) Value of SONATA Fusion.
The fusion of 2D visual features with 3D geometric descriptors using SONATA contributed to more robust pose estimation. Compared to DINO-only matching pipelines like DINO-Patches, the proposed hybrid model achieved reduced yaw error and showed more stable behavior in low-texture regions. This demonstrates the benefit of combining multiple sensing modalities in environments where purely visual information is unreliable or ambiguous.

e) Understanding Reloc3r's Advantage.
While the regression-based Reloc3r model lacks interpretability, its high accuracy and speed are due to direct end-to-end optimization. By minimizing pose error during training, it avoids the need for matching or explicit geometry, making it less sensitive to feature quality or scene overlap. However, its black-box nature and potential failure under distribution shifts make it less suitable for high-stakes applications where diagnosis and robustness are critical.

In conclusion, these findings reinforce the importance of domain-specific fine-tuning, careful feature selection, and hybrid architectures in building reliable

localization pipelines. While end-to-end systems like Reloc3r offer strong performance, matching-based pipelines with interpretable modules, especially when enhanced by multi-modal fusion, remain robust and transparent in the types of environments targeted in this work.

6.4 Unexpected Observations

While the experimental results largely aligned with expectations, several findings stood out as surprising or counterintuitive. These observations offer deeper insight into the limitations and behaviors of the evaluated methods:

a) **Underperformance of DINO-Patch Matching.** Despite leveraging strong visual features, the DINO-Patches baseline delivered less robust pose estimates than anticipated. This may be due to coarse spatial resolution or suboptimal patch alignment, which led to inconsistent match quality in low-texture regions. The lack of geometric priors also made it more vulnerable to perceptual aliasing.

b) **FPFH's Mixed Practicality.** Although FPFH was expected to fail completely in unstructured terrain, it still produced valid poses in a surprising number of cases. However, its computational cost was prohibitively high, and the pose accuracy was inconsistent. This confirms that while handcrafted features can sometimes succeed, they are not scalable or dependable for real-time deployment.

c) **LoFTR's Translation–Yaw Discrepancy.** LoFTR achieved strong translation accuracy but showed poor stability in yaw estimation. This suggests that while dense correspondence fields may be sufficient for estimating position, they are more susceptible to rotational drift, possibly due to ambiguous feature orientations or local symmetries.

d) **Performance Gaps Between Similar DINO Variants.** The large performance jump between DINOv2 (b) using CLS tokens vs. the last three layers was larger than expected. This highlights the sensitivity of self-supervised descriptors to subtle architectural and pooling choices, and the importance of layer selection in transfer learning.

6.5 Impact and Practical Relevance

The method proposed in this thesis addresses a key gap in the visual localization literature by offering a pipeline that delivers both *image retrieval* and *pose estimation*—a capability rarely achieved in a single system. While many existing approaches focus exclusively on retrieval for place recognition or regression-based pose estimation, few provide a full pipeline that is modular, interpretable, and robust across both tasks.

This work introduces a matching-based architecture that combines 2D self-supervised features with 3D geometric descriptors, striking a valuable balance between **interpretability**, **modularity**, and **robustness**. These properties are essential in safety-critical scenarios such as planetary exploration or disaster response, where explainability and diagnostic access are nonnegotiable.

Equally important is the consideration of **computational efficiency**, especially for deployment in real-time autonomous systems. Although the current pose estimation module is relatively slow (3114.33 ms per query), the image retrieval component operates at 476.5 ms and is therefore compatible with real-time use on embedded platforms. Optimizing or approximating the pose estimation stage remains a promising direction for future work.

This makes the system particularly well-suited for field robotics in GNSS-denied and unstructured environments, such as planetary exploration, disaster response, or agricultural monitoring—scenarios where semantic structure is limited, connectivity may be absent, and runtime constraints are strict.

The key takeaway is that **domain-adapted, multi-modal fusion pipelines** can offer strong performance without sacrificing explainability. Future systems can build upon this work by extending multi-modal integration, improving runtime for onboard execution, or adding uncertainty quantification to support decision-making in critical missions.

Chapter 7

Conclusions and Future work

This chapter summarizes the key contributions, insights, and outcomes of the thesis. Section 7.1 revisits the three main objectives, detailing how each was addressed through system design, experimentation, and evaluation. Section 7.2 outlines the key limitations encountered during the project, particularly in terms of generalizability, computational cost, and evaluation scope. Section 7.3 proposes future research directions to improve performance, expand applicability, and support real-world deployment. Finally, Section 7.4 reflects on the ethical, societal, and environmental implications of the work, aligning the system's design with broader goals such as sustainability, safety, and transparency in robotics.

7.1 Conclusions

This thesis aimed to develop a multi-modal place recognition and pose estimation system for unstructured, GNSS-denied environments, focusing on both accuracy and interpretability. The motivation stemmed from the lack of existing methods capable of delivering 6-DoF pose estimates—rather than simple retrieval—under visually sparse and ambiguous conditions. The targeted use case was integration with SLAM pipelines in resource-constrained robots like the LRU, making efficiency and modularity essential considerations.

To meet this goal, the work was structured around three main objectives:

This thesis aimed to develop a multi-modal place recognition and pose estimation system tailored for unstructured, GNSS-denied environments, with a focus on producing interpretable, SLAM-compatible 6D pose outputs. The system was designed to balance accuracy, robustness, and efficiency for use

in field robotics platforms such as the LRU. The work was structured around three main objectives:

Objective 1: Evaluate the benefits of multi-modality (vision + LiDAR) in place recognition and pose estimation. This objective was fully achieved and is supported by results throughout the thesis.

- **Literature Review:** Chapter 2 provides an extensive overview of state-of-the-art multi-modal learning techniques and identifies limitations in existing approaches, especially under unstructured terrain conditions.
- **Data Preprocessing:** Chapter 3.2 and 4.1 detail the processing of the Etna dataset, chosen for its resemblance to planetary environments. LiDAR and camera synchronization was performed using the S3LI toolkit to ensure aligned multi-modal input [90].
- **Baseline Analysis:** While the implementation of prior multi-modal frameworks (e.g., MinkLoc, UMF, AdaFusion) was hindered by technical and reproducibility issues, this thesis demonstrated through ablation that the proposed fusion of DINOv2 (vision) and SONATA (LiDAR) outperformed their standalone counterparts [34, 73]. This validates the hypothesis that combining modalities improves pose robustness, especially in texture-poor scenes.

Objective 2: Develop an algorithm that outputs 6D poses for SLAM integration instead of simple image retrieval. This objective was successfully met.

- **Feature Extraction and Fusion:** The system uses transformer-based features from DINOv2 and SONATA, fused by projecting 2D visual features into 3D space before concatenation. Alternative fusion techniques, such as weighted averaging, were explored but found less effective.
- **Pose Estimation:** The final pipeline outputs a 6-DoF transformation matrix, enabling SLAM integration. While evaluation was restricted to yaw and translation (due to dataset constraints), the full output structure satisfies the original goal.
- **Benchmarking:** Metrics and baseline comparisons are presented in Chapters 3.3 and 5. The proposed method achieved competitive

accuracy and robustness when compared with regression-based and traditional feature-based pose estimators.

Objective 3: Optimize computational and memory efficiency by integrating model-based components into deep learning-based multi-modal methods. This objective was partially achieved, although it revealed important trade-offs between transparency, runtime, and robustness.

- **Model-Based Integration:** Classical geometric modules such as RANSAC were incorporated into the pose estimation pipeline to infer transformations from matched features [16]. This contributed to interpretability and reduced the reliance on fully end-to-end learned pose estimators.
- **Efficient Inference:** The image retrieval pipeline was optimized using descriptor aggregation via SALAD and fast similarity search with FAISS indexing [26, 27], resulting in a mean inference time of under 500 ms—suitable for real-time deployment. Pose estimation, on the other hand, remained computationally heavy (approx. 3.1 seconds), which limits immediate deployment but provides a transparent and modular structure that is open to future optimization.
- **Hardware Constraints:** Runtime measurements reflect realistic GPU settings. Although SONATA could not be fine-tuned due to hardware limitations, its pre-trained performance was strong, and the system remains adaptable for constrained platforms.

From a broader perspective, the project confirmed that hybrid pipelines combining interpretable, modular components with domain-adapted features can deliver competitive performance even in the absence of structured semantics.

Some challenges and trade-offs emerged. Re-training SALAD after changing DINO features led to degraded performance, underscoring the fragility of VLAD-based learning. Similarly, while the proposed matching-based pose estimator offered strong interpretability and robustness, its computational cost limits its current applicability in real-time systems. Nevertheless, the pipeline offers a valuable foundation for future SLAM systems that require both accurate retrieval and metric pose estimation, a combination still underexplored in the literature.

Key insights gained include:

- The importance of fine-tuning, even for powerful self-supervised models like DINOv2, which showed a 4.5
- The value of fusing 2D and 3D modalities, particularly in texture-poor scenes.
- The potential of regression models like Reloc3r, which, despite lacking interpretability, delivered strong pose accuracy, suggesting room for hybridization with interpretable pipelines.

If this project were to be repeated, more focus would be placed on exploring regressors for pose estimation, and with more computational resources, it would have been feasible to fine-tune geometric backbones like SONATA, potentially unlocking further performance gains.

To conclude, this work demonstrates that multi-modal, interpretable localization systems can achieve reliable performance in visually ambiguous, unstructured environments, while also paving the way for future deployment in SLAM for planetary robotics. The system fills a niche in the current research space by addressing both retrieval and pose estimation with a balance of explainability, modularity, and performance.

7.2 Limitations

While the proposed multi-modal localization pipeline achieved promising results, several limitations affect its generalizability, efficiency, and applicability.

Single-Dataset Evaluation. All experiments were conducted exclusively on the Etna dataset [21]. Although this dataset provides a suitable approximation of planetary terrain, it does not capture the full variability of real-world or extraterrestrial conditions. Additional datasets, including those with different terrain types, weather conditions, or sensor configurations, would be required to validate the method's robustness more comprehensively.

Incomplete 6D Evaluation. Although the system is designed to output full 6-DoF pose transformations, quantitative evaluation was restricted to yaw and planar translation. This was due to the nature of the dataset, where variations in roll, pitch, and vertical displacement were minimal and could not be reliably assessed. As a result, further testing in more dynamic environments is necessary to confirm the accuracy of the full 6D output.

Pose Estimation Runtime. Despite the real-time performance of the image retrieval module (476.5 ms), the pose estimation module remains computationally heavy, averaging over 3 seconds per query. This runtime is too high for seamless real-time deployment in SLAM pipelines, particularly in time-sensitive robotic applications. Additional engineering work is needed to optimize this component.

Unrealized Multi-Modal Baselines. Due to time and compatibility constraints, state-of-the-art multi-modal baselines such as AdaFusion, MinkLoc, and UMF could not be successfully evaluated. Their absence limits the breadth of comparative analysis, although alternative single-modality and hybrid baselines were included for benchmarking.

Hardware Constraints. The inability to fine-tune SONATA was a direct result of limited computational resources. While the backbone performed well using pre-trained weights, fine-tuning could have yielded even better domain adaptation and feature alignment.

7.3 Future work

The work presented in this thesis opens several directions for continued research and development:

Fine-Tuning SONATA. With sufficient computational resources, future work should focus on fine-tuning SONATA to better adapt geometric features to unstructured, planetary-like terrain. This could reduce pose error and improve match robustness, especially in sparse or noisy point clouds.

Semantic-Aware Pose Refinement. A promising extension would involve integrating semantic information into the geometric alignment process, e.g., enhancing ICP refinement by minimizing distances not only between point positions but also point-wise semantic features. This could reduce the ambiguity of low-texture or repetitive surfaces and improve registration accuracy.

Speed Optimization of Pose Estimation. Significant engineering work remains to reduce the computational cost of the pose estimation module.

Options include distillation of feature extractors, faster match pruning strategies, or lightweight regressors that operate after an initial matching stage.

Robust SLAM Integration and Field Testing. Future iterations of the system should be deployed on robotic hardware such as the LRU for real-time testing [19]. This would allow validation of the system's SLAM compatibility, feedback-loop behavior, and error recovery in operational scenarios.

Dataset Expansion. New datasets, such as the S3LI Vulcano dataset, could be used to evaluate the generalization of the pipeline to novel terrain and sensor conditions. Additionally, incorporating real-world data from planetary exploration robotics would enhance relevance and robustness.

Extended Baseline Comparisons. Revisiting multi-modal methods like UMF, MinkLoc, and AdaFusion, as well as LiDAR-only approaches like PointNetVLAD, would allow for a more complete benchmarking of the proposed system. Overcoming reproducibility issues in these baselines remains a useful effort for the research community.

Alternative Pose Estimators. Given the strong performance of regression-based pose models such as Reloc3r [63], future work could explore hybrid models that retain interpretability through initial matching but refine outputs using compact pose regressors. This could improve the speed-accuracy trade-off while maintaining transparency.

7.4 Reflections

This section provides a broader reflection on the ethical, environmental, and societal aspects of the work, in line with the United Nations (UN) Sustainable Development Goalss (SDGs).

Environmental Sustainability (SDG 13: Climate Action). From a sustainability standpoint, the focus on computational efficiency directly contributes to reducing the energy footprint of autonomous systems. By employing compact image descriptors (e.g., SALAD [26]), efficient search indexing (via FAISS [27]), and modular architecture, the proposed system avoids the heavy resource consumption typical of end-to-end models. Moreover, real-time onboard processing reduces reliance on remote servers,

making the method suitable for deployment in low-connectivity environments with limited infrastructure.

Autonomous systems that rely on visual-LiDAR SLAM are increasingly used in sustainable agriculture (e.g., precision weeding, soil monitoring), where they enable optimized use of resources like water, fertilizer, or fuel. By improving navigation robustness in natural terrain, the methods developed here can help scale up these applications.

Social Relevance (SDG 9: Industry, Innovation, and Infrastructure). The system presented in this thesis has potential applications in planetary exploration, disaster response, environmental monitoring, and autonomous field robotics. These use cases are often safety-critical, occurring in GPS-denied, unpredictable environments where human operation is infeasible. Providing reliable and interpretable localization under such conditions can facilitate faster rescue missions, advance space research, and support critical infrastructure inspection.

Furthermore, enabling affordable and accurate autonomous navigation contributes to democratizing access to robotics technologies in under-resourced regions or industries, fostering inclusive technological development.

Ethical Considerations. Interpretability and modularity were central design principles of the proposed system. Unlike black-box regressors, matching-based pipelines allow users to trace the reasoning behind localization estimates—an essential property for safety-critical deployments. This aligns with growing ethical requirements around explainability and accountability in AI and robotics.

From a data ethics perspective, the use of public and non-sensitive datasets (e.g., Etna [21]) ensures that the development process avoids privacy violations or misuse. However, broader deployment, especially in surveillance or defense contexts, requires ongoing vigilance regarding the potential for unintended applications or societal harm.

References

[1] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard, "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," *IEEE Transactions on Robotics*, vol. 32, no. 6, pp. 1309–1332, 12 2016. doi: 10.1109/TRO.2016.2624754 [Pages 1, 13, and 14.]

[2] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual-Inertial and Multi-Map SLAM," *IEEE Transactions on Robotics*, vol. 37, no. 6, pp. 1874–1890, 7 2020. doi: 10.1109/TRO.2021.3075644. [Online]. Available: http://arxiv.org/abs/2007.11898http://dx.doi.org/10.1109/TRO.2021.3075644 [Pages 1 and 16.]

[3] D. Gálvez-López and J. D. Tardós, "Bags of binary words for fast place recognition in image sequences," *IEEE Transactions on Robotics*, vol. 28, no. 5, pp. 1188–1197, 2012. doi: 10.1109/TRO.2012.2197158 [Pages 2 and 22.]

[4] M. Muja and D. G. Lowe, "Fast Approximate Nearest Neighbors With Automatic Algorithm Configuration." [Page 2.]

[5] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 11 2004. doi: 10.1023/B:VISI.0000029664.99615.94 [Pages 2, 15, and 16.]

[6] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2564–2571, 2011. doi: 10.1109/ICCV.2011.6126544 [Pages 2, 16, and 35.]

[7] D. Detone, T. Malisiewicz, and A. Rabinovich, "SuperPoint: Self-Supervised Interest Point Detection and Description." [Page 2.]

[8] J. Sun, Z. Shen, Y. Wang, H. Bao, and X. Zhou, "LoFTR: Detector-Free Local Feature Matching with Transformers," Tech. Rep., 4 2021. [Online]. Available: https://zju3dv.github.io/loftr/. [Pages 2, 17, 23, 49, 78, and 79.]

[9] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "NetVLAD: CNN architecture for weakly supervised place recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 6, pp. 1437–1451, 11 2015. doi: 10.1109/TPAMI.2017.2711011. [Online]. Available: https://arxiv.org/abs/1511.07247v3 [Pages 2, 16, 19, 30, 39, 49, 66, 76, 77, and 78.]

[10] P. E. Sarlin, C. Cadena, R. Siegwart, and M. Dymczyk, "From Coarse to Fine: Robust Hierarchical Localization at Large Scale," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2019-June, pp. 12 708–12 717, 12 2018. doi: 10.1109/CVPR.2019.01300. [Online]. Available: https://arxiv.org/abs/1812.03506v2 [Pages 2, 23, and 34.]

[11] A. Garcia-Hernandez, R. Giubilato, K. H. Strobl, J. Civera, and R. Triebel, "Unifying Local and Global Multimodal Features for Place Recognition in Aliased and Low-Texture Environments," *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 3991–3998, 3 2024. doi: 10.1109/icra57147.2024.10611563. [Online]. Available: https://arxiv.org/abs/2403.13395v1 [Pages 2, 3, 25, 30, 36, and 49.]

[12] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation," *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, vol. 2017-January, pp. 77–85, 12 2016. doi: 10.1109/CVPR.2017.16. [Online]. Available: https://arxiv.org/abs/1612.00593v2 [Pages 2 and 30.]

[13] H. Lai, P. Yin, and S. Scherer, "AdaFusion: Visual-LiDAR Fusion with Adaptive Weights for Place Recognition," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 12 038–12 045, 11 2021. doi: 10.1109/LRA.2022.3210880. [Online]. Available: https://arxiv.org/abs/2111.11739v1 [Pages 2, 30, and 49.]

[14] A. Kendall, M. Grimes, and R. Cipolla, "PoseNet: A Convolutional Network for Real-Time 6-DOF Camera Relocalization," 5 2015. [Online]. Available: http://arxiv.org/abs/1505.07427 [Pages 2 and 27.]

[15] X. S. Gao, X. R. Hou, J. Tang, and H. F. Cheng, "Complete solution classification for the perspective-three-point problem," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 8, pp. 930–943, 8 2003. doi: 10.1109/TPAMI.2003.1217599. [Online]. Available: https://www.researchgate.net/publication/3193582_Complete_Solution_Classification_for_the_Perspective-Three-Point_Problem [Pages 2, 25, and 26.]

[16] M. A. Fischler and R. C. Bolles, "Random sample consensus," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 6 1981. doi: 10.1145/358669.358692. [Online]. Available: https://dl.acm.org/doi/10.1145/358669.358692 [Pages 2, 25, 32, 35, 42, 49, 79, and 85.]

[17] "COLMAP — COLMAP 3.12.0.dev0 documentation." [Online]. Available: https://colmap.github.io/ [Pages 2, 26, and 29.]

[18] J. Zhang, Y. Yao, and B. Deng, "Fast and Robust Iterative Closest Point," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 7, pp. 3450–3466, 7 2020. doi: 10.1109/tpami.2021.3054619. [Online]. Available: https://arxiv.org/abs/2007.07627v3 [Pages 2, 26, 32, and 42.]

[19] "LRU." [Online]. Available: https://www.dlr.de/en/rm/research/robotic-systems/mobile-platforms/lru [Pages 3, 4, 5, and 88.]

[20] "The German Aerospace Center (DLR)." [Online]. Available: https://www.dlr.de/en [Page 5.]

[21] R. Giubilato, W. Sturzl, A. Wedler, and R. Triebel, "Challenges of SLAM in Extremely Unstructured Environments: The DLR Planetary Stereo, Solid-State LiDAR, Inertial Dataset," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 8721–8728, 10 2022. doi: 10.1109/LRA.2022.3188118 [Pages 8, 9, 44, 47, 54, 86, and 89.]

[22] L. Meyer, M. Smíšek, A. Fontan Villacampa, L. Oliva Maza, D. Medina, M. J. Schuster, F. Steidle, M. Vayugundla, M. G. Müller, B. Rebele, A. Wedler, and R. Triebel, "The MADMAX data set for visual-inertial

rover navigation on Mars," *Journal of Field Robotics*, vol. 38, no. 6, pp. 833–853, 9 2021. doi: 10.1002/rob.22016 [Page 8.]

[23] F. Warburg, S. Hauberg, M. López-Antequera, P. Gargallo, Y. Kuang, and J. Civera, "Mapillary street-level sequences: A dataset for lifelong place recognition," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 2623–2632, 2020. doi: 10.1109/CVPR42600.2020.00270 [Pages 9 and 48.]

[24] A. Ali-bey, B. Chaib-draa, and P. Giguère, "GSV-Cities: Toward Appropriate Supervised Visual Place Recognition," *Neurocomputing*, vol. 513, pp. 194–203, 10 2022. doi: 10.1016/j.neucom.2022.09.127. [Online]. Available: http://arxiv.org/abs/2210.10239http://dx.doi.org/10.1016/j.neucom.2022.09.127 [Pages 9 and 48.]

[25] A. Torii, J. Sivic, T. Pajdla, and M. Okutomi, "Visual place recognition with repetitive structures," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 883–890, 2013. doi: 10.1109/CVPR.2013.119 [Pages 9 and 48.]

[26] S. Izquierdo and J. Civera, "Optimal Transport Aggregation for Visual Place Recognition," 11 2023. doi: 10.1109/CVPR52733.2024.01672. [Online]. Available: https://arxiv.org/abs/2311.15937v2 [Pages 10, 19, 31, 34, 37, 39, 40, 48, 49, 66, 77, 78, 85, and 88.]

[27] "Faiss." [Online]. Available: https://ai.meta.com/tools/faiss/ [Pages 10, 21, 34, 39, 59, 85, and 88.]

[28] S. Schubert, S. Garg, M. Milford, and T. Fischer, "Visual Place Recognition: A Tutorial," 8 2023. [Online]. Available: https://github.com/ [Page 14.]

[29] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded up robust features," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 3951 LNCS, pp. 404–417, 2006. doi: 10.1007/11744023_32 [Pages 16 and 35.]

[30] T. Barros, R. Pereira, L. Garrote, C. Premebida, and U. J. Nunes, "Place recognition survey: An update on deep learning approaches," 6 2021. [Online]. Available: https://arxiv.org/abs/2106.10458v3 [Pages 16, 23, and 37.]

[31] F. Magliani, T. Fontanini, and A. Prati, "A Dense-Depth Representation for VLAD descriptors in Content-Based Image Retrieval," Tech. Rep., 8 2018. [Online]. Available: http://implab.ce.unipr.it [Page 16.]

[32] S. Hausler, S. Garg, M. Xu, M. Milford, and T. Fischer, "Patch-NetVLAD: Multi-Scale Fusion of Locally-Global Descriptors for Place Recognition," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2021)*, 2021. [Online]. Available: http://ieeexplore.ieee.org [Pages 16, 18, and 19.]

[33] M. Caron, H. Touvron, I. Misra, H. Jegou, J. Mairal, P. Bojanowski, and A. Joulin, "Emerging Properties in Self-Supervised Vision Transformers," *Proceedings of the IEEE International Conference on Computer Vision*, pp. 9630–9640, 4 2021. doi: 10.1109/ICCV48922.2021.00951. [Online]. Available: https://arxiv.org/abs/2104.14294v2 [Pages 17 and 18.]

[34] M. Oquab, T. Darcet, T. Moutakanni, H. V. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, M. Assran, N. Ballas, W. Galuba, R. Howes, P.-Y. Huang, S.-W. Li, I. Misra, M. Rabbat, V. Sharma, G. Synnaeve, H. Xu, H. Jegou, J. Mairal, P. Labatut, A. Joulin, and P. Bojanowski, "DINOv2: Learning Robust Visual Features without Supervision," 4 2023. [Online]. Available: https://arxiv.org/abs/2304.07193v2 [Pages 17, 18, 31, 34, 35, 37, 49, 60, 66, 77, and 84.]

[35] N. Keetha, A. Mishra, J. Karhade, K. M. Jatavallabhula, S. Scherer, M. Krishna, and S. Garg, "AnyLoc: Towards Universal Visual Place Recognition," *IEEE Robotics and Automation Letters*, vol. 9, no. 2, pp. 1286–1293, 8 2023. doi: 10.1109/LRA.2023.3343602. [Online]. Available: https://arxiv.org/abs/2308.00688v2 [Pages 18 and 19.]

[36] R. Wang, Y. Shen, W. Zuo, S. Zhou, and N. Zheng, "TransVPR: Transformer-based place recognition with multi-level attention aggregation," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2022-June, pp. 13 638–13 647, 1 2022. doi: 10.1109/CVPR52688.2022.01328. [Online]. Available: https://arxiv.org/abs/2201.02001v4 [Pages 18, 66, 76, 77, and 78.]

[37] I. Tolstikhin, N. Houlsby, A. Kolesnikov, L. Beyer, X. Zhai, T. Unterthiner, J. Yung, A. Steiner, D. Keysers, J. Uszkoreit, M. Lucic,

and A. Dosovitskiy, "MLP-Mixer: An all-MLP Architecture for Vision," *Advances in Neural Information Processing Systems*, vol. 29, pp. 24 261–24 272, 5 2021. [Online]. Available: https://arxiv.org/abs/2105.01601v4 [Page 18.]

[38] G. Huang, Y. Zhou, X. Hu, C. Zhang, L. Zhao, W. Gan, and M. Hou, "DINO-Mix: Enhancing Visual Place Recognition with Foundational Vision Model and Feature Mixing," 11 2023. [Online]. Available: https://arxiv.org/abs/2311.00230v2 [Pages 18 and 37.]

[39] A. Ali-Bey, B. Chaib-Draa, and P. Giguere, "MixVPR: Feature Mixing for Visual Place Recognition," *Proceedings - 2023 IEEE Winter Conference on Applications of Computer Vision, WACV 2023*, pp. 2997–3006, 3 2023. doi: 10.1109/WACV56688.2023.00301. [Online]. Available: https://arxiv.org/abs/2303.02190v1 [Pages 18 and 20.]

[40] P. E. Sarlin, D. Detone, T. Malisiewicz, and A. Rabinovich, "SuperGlue: Learning Feature Matching with Graph Neural Networks," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 4937–4946, 2020. doi: 10.1109/CVPR42600.2020.00499 [Pages 18 and 23.]

[41] L. Chirca, "Enhancing Sequential Visual Place Recognition With Foundational Vision Model and Spatio-Temporal Feature mixing," 8 2024. [Page 19.]

[42] "Global Average Pooling Explained | Papers With Code." [Online]. Available: https://paperswithcode.com/method/global-average-pooling [Page 19.]

[43] "Pooling layer - Wikipedia." [Online]. Available: https://en.wikipedia.org/wiki/Pooling_layer [Page 19.]

[44] Y. Gu, C. Li, and J. Xie, "Attention-Aware Generalized Mean Pooling for Image Retrieval," 11 2018. [Online]. Available: https://arxiv.org/abs/1811.00202v2 [Pages 19, 20, and 39.]

[45] B. Ko, H.-G. Kim, B. Heo, S. Yun, S. Chun, G. Gu, and W. Kim, "Group Generalized Mean Pooling for Vision Transformer," 12 2022. [Online]. Available: https://arxiv.org/abs/2212.04114v1 [Pages 19 and 39.]

[46] Z. Wang, W. Di, A. Bhardwaj, V. Jagadeesh, and R. Piramuthu, "Geometric VLAD for Large Scale Image Search," 3 2014. [Online]. Available: https://arxiv.org/abs/1403.3829v1 [Page 19.]

[47] H. Kristín Ólafsdóttir, H. Rootzén, and D. Bolin, "Fast and robust cross-validation-based scoring rule inference for spatial statistics." [Pages 19 and 25.]

[48] M. Cuturi, "Sinkhorn Distances: Lightspeed Computation of Optimal Transport," *Advances in Neural Information Processing Systems*, vol. 26, 2013. [Page 20.]

[49] R. Zhu, "Fast Exact Retrieval for Nearest-neighbor Lookup (FERN)," 5 2024. [Online]. Available: http://arxiv.org/abs/2405.04435 [Page 20.]

[50] M. Greenspan and M. Yurick, "Approximate k-d tree search for efficient ICP," *Proceedings of International Conference on 3-D Digital Imaging and Modeling, 3DIM*, vol. 2003-January, pp. 442–448, 2003. doi: 10.1109/IM.2003.1240280 [Page 21.]

[51] J. Engels, B. Landrum, S. Yu, L. Dhulipala, and J. Shun, "Approximate Nearest Neighbor Search with Window Filters," 2 2024. [Online]. Available: https://arxiv.org/abs/2402.00943v2 [Page 21.]

[52] S. Gidaris, A. Bursuc, N. Komodakis, P. Perez, and M. Cord, "Learning Representations by Predicting Bags of Visual Words," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 6926–6936, 2 2020. doi: 10.1109/CVPR42600.2020.00696. [Online]. Available: https://arxiv.org/abs/2002.12247v1 [Page 21.]

[53] Z. Tan, H. Wang, B. Xu, M. Luo, and M. Du, "Fast Locality Sensitive Hashing with Theoretical Guarantee," 9 2023. [Online]. Available: https://arxiv.org/abs/2309.15479v1 [Page 22.]

[54] Y. Weiss, A. Torralba, and R. Fergus, "Spectral Hashing," *Advances in Neural Information Processing Systems*, vol. 21, 2008. [Page 22.]

[55] "OpenCV: Feature Matching with FLANN." [Online]. Available: https://docs.opencv.org/3.4/d5/d6f/tutorial_feature_flann_matcher.html [Page 23.]

[56] Z. Li and N. Snavely, "MegaDepth: Learning Single-View Depth Prediction from Internet Photos," 11 2018. [Online]. Available: http://www.cs.cornell.edu/projects/ [Page 24.]

[57] F. Lu, S. Dong, L. Zhang, B. Liu, X. Lan, D. Jiang, and C. Yuan, "Deep Homography Estimation for Visual Place Recognition," 2024. [Online]. Available: https://github.com/Lu-Feng/DHE-VPR. [Page 24.]

[58] Y. Hao, M. He, Y. Liu, J. Liu, and Z. Meng, "Range–Visual–Inertial Odometry with Coarse-to-Fine Image Registration Fusion for UAV Localization," *Drones*, vol. 7, no. 8, 8 2023. doi: 10.3390/DRONES7080540 [Page 25.]

[59] E. Brachmann, A. Krull, S. Nowozin, J. Shotton, F. Michel, S. Gumhold, and C. Rother, "DSAC - Differentiable RANSAC for Camera Localization," *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, vol. 2017-January, pp. 2492–2500, 11 2016. doi: 10.1109/CVPR.2017.267. [Online]. Available: https://arxiv.org/abs/1611.05705v4 [Pages 26 and 28.]

[60] E. P. Örnek, Y. Labbé, B. Tekin, L. Ma, C. Keskin, C. Forster, and T. Hodan, "FoundPose: Unseen Object Pose Estimation with Foundation Features," 11 2023. doi: 10.1007/978-3-031-73347-5_10. [Online]. Available: https://arxiv.org/abs/2311.18809v2 [Pages 26, 29, 49, 78, and 79.]

[61] "resnet34 — Torchvision main documentation." [Online]. Available: https://docs.pytorch.org/vision/main/models/generated/torchvision.models.resnet34.html [Page 27.]

[62] F. Khatib, Y. Margalit, M. Galun, and R. Basri, "Leveraging Image Matching Toward End-to-End Relative Camera Pose Regression," 11 2022. [Online]. Available: https://arxiv.org/abs/2211.14950v2 [Pages 27 and 29.]

[63] S. Dong, S. Wang, S. Liu, L. Cai, Q. Fan, J. Kannala, and Y. Yang, "Reloc3r: Large-Scale Training of Relative Camera Pose Regression for Generalizable, Fast, and Accurate Visual Localization," 12 2024. [Online]. Available: https://arxiv.org/abs/2412.08376v2 [Pages 27, 49, 76, 78, 79, and 88.]

[64] C. Rockwell, J. Johnson, and D. F. Fouhey, "The 8-Point Algorithm as an Inductive Bias for Relative Pose Prediction by ViTs," *Proceedings - 2022 International Conference on 3D Vision, 3DV 2022*, pp. 155–165, 8 2022. doi: 10.1109/3DV57658.2022.00028. [Online]. Available: https://arxiv.org/abs/2208.08988v2 [Page 28.]

[65] "Matterport3D Dataset | Papers With Code." [Online]. Available: https://paperswithcode.com/dataset/matterport3d [Page 28.]

[66] M. A. Uy and G. H. Lee, "PointNetVLAD: Deep Point Cloud Based Retrieval for Large-Scale Place Recognition," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 4470–4479, 4 2018. doi: 10.1109/CVPR.2018.00470. [Online]. Available: https://arxiv.org/abs/1804.03492v3 [Pages 30 and 49.]

[67] Z. Li, T. Shang, P. Xu, and Z. Deng, "Place Recognition Meet Multiple Modalities: A Comprehensive Review, Current Challenges and Future Development." [Online]. Available: https://github.com/CV4RA/SOTA-Place- [Page 30.]

[68] J. Komorowski, M. Wysoczanska, and T. Trzcinski, "MinkLoc++: Lidar and Monocular Image Fusion for Place Recognition," *Proceedings of the International Joint Conference on Neural Networks*, vol. 2021-July, 4 2021. doi: 10.1109/IJCNN52387.2021.9533373. [Online]. Available: https://arxiv.org/abs/2104.05327v2 [Pages 30 and 49.]

[69] W. Maddern, G. Pascoe, C. Linegar, and P. Newman, "1 year, 1000 km: The Oxford RobotCar dataset," *International Journal of Robotics Research*, vol. 36, no. 1, pp. 3–15, 1 2017. doi: 10.1177/0278364916679498. [Online]. Available: https://robotcar-dataset.robots.ox.ac.uk/citation/ [Page 31.]

[70] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets Robotics: The KITTI Dataset." [Online]. Available: http://www.cvlibs.net/datasets/kitti. [Page 31.]

[71] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked Autoencoders Are Scalable Vision Learners," Tech. Rep. [Page 31.]

[72] H. Chen, Z. Luo, L. Zhou, Y. Tian, M. Zhen, T. Fang, D. Mckinnon, Y. Tsin, and L. Quan, "ASpanFormer: Detector-Free Image Matching with Adaptive Span Transformer," Tech. Rep. [Page 31.]

[73] X. Wu, D. Detone, D. Frost, T. Shen, C. Xie, N. Yang, J. Engel, R. Newcombe, H. Zhao, and J. Straub, "Sonata: Self-Supervised Learning of Reliable Point Representations Semantic Awareness Perception Self-distillation PCA K-means Dense Matching Sparse Matching," Tech. Rep., 2025. [Online]. Available: https://github.com/facebookresearch/sonata [Pages 31, 40, 49, 60, and 84.]

[74] Y. Zheng, G. Wang, J. Liu, M. Pollefeys, and H. Wang, "Spherical Frustum Sparse Convolution Network for LiDAR Point Cloud Semantic Segmentation," Tech. Rep. [Online]. Available: https://github.com/IRMVLab/SFCNet. [Page 32.]

[75] P. Biasutti, V. Lepetit, M. Brédif, J.-F. Aujol, and A. Bugeau, "LU-Net: An Efficient Network for 3D LiDAR Point Cloud Semantic Segmentation Based on End-to-End-Learned 3D Features and U-Net," Tech. Rep. [Page 32.]

[76] Y. Wang, Y. Dai, Q. Liu, P. Yang, J. Sun, and B. Li, "CU-Net: LiDAR Depth-only Completion with Coupled U-Net," Tech. Rep. [Online]. Available: https://github.com/YufeiWang777/CU-Net. [Page 32.]

[77] R. B. Rusu, N. Blodow, and M. Beetz, "Fast Point Feature Histograms (FPFH) for 3D Registration," *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 3212–3217, 2009. doi: 10.1109/ROBOT.2009.5152473 [Pages 32, 49, 76, and 79.]

[78] A. Zeng, S. Song, M. Nießner, M. Fisher, J. Xiao, and T. Funkhouser, "3DMatch: Learning Local Geometric Descriptors from RGB-D Reconstructions," Tech. Rep. [Online]. Available: http://3dmatch.cs.princeton.eduhttp://3dmatch.cs.princeton.edu. [Page 32.]

[79] N. Vödisch, G. Cioffi, M. Cannici, W. Burgard, and D. Scaramuzza, "LiDAR Registration with Visual Foundation Models," Tech. Rep. [Online]. Available: https://vfm-registration.cs.uni-freiburg.de. [Pages 32, 42, 49, and 79.]

[80] G. Pramatarov, D. De Martini, M. Gadd, and P. Newman, "BoxGraph: Semantic Place Recognition and Pose Estimation from 3D LiDAR," *IEEE International Conference on Intelligent Robots and Systems*, vol. 2022-October, pp. 7004–7011, 6 2022. doi: 10.1109/IROS47612.2022.9981266. [Online]. Available: https://arxiv.org/abs/2206.15154v1 [Page 35.]

[81] S. Lu, X. Xu, H. Yin, Z. Chen, R. Xiong, and Y. Wang, "One RING to Rule Them All: Radon Sinogram for Place Recognition, Orientation and Translation Estimation," Tech. Rep. [Page 36.]

[82] H. Li, C. Sima, J. Dai, W. Wang, L. Lu, H. Wang, J. Zeng, Z. Li, J. Yang, H. Deng, H. Tian, E. Xie, J. Xie, L. Chen, T. Li, Y. Li, Y. Gao, X. Jia, S. Liu, J. Shi, D. Lin, and Y. Qiao, "Delving into the Devils of Bird's-eye-view Perception: A Review, Evaluation and Recipe." [Online]. Available: https://github.com/OpenDriveLab/Birds-eye-view-Perception. [Page 36.]

[83] H. Jang, M. Jung, and A. Kim, "RaPlace: Place Recognition for Imaging Radar using Radon Transform and Mutable Threshold," Tech. Rep. [Online]. Available: https://github.com/hyesu-jang/RaPlace. [Page 36.]

[84] D. Jung, K. Kim, and S.-W. Kim, "GOTPR: General Outdoor Text-based Place Recognition Using Scene Graph Retrieval with OpenStreetMap," *IEEE ROBOTICS AND AUTOMATION LETTERS*, vol. 10, no. 6, 2025. doi: 10.1109/LRA.2025.3568306. [Online]. Available: https://doi.org/10.1109/LRA.2025.3568306, [Page 39.]

[85] T. Ye, A. Liu, X. Yan, X. Yan, Y. Ouyang, X. Deng, X. Cong, and F. Zhang, "An Efficient 3D Point Cloud-Based Place Recognition Approach for Underground Tunnels Using Convolution and Self-Attention Mechanism," *Journal of Field Robotics*, vol. 42, no. 4, pp. 1537–1549, 6 2025. doi: 10.1002/ROB.22451. [Online]. Available: https://onlinelibrary.wiley.com/doi/full/10.1002/rob.22451https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.22451https://onlinelibrary.wiley.com/doi/10.1002/rob.22451 [Page 39.]

[86] P. K. Rai and R. Ghabcheloo, "Representation Learning for Place Recognition Using MIMO Radar." doi: 10.1109/OJITS.2025.3543286 [Page 39.]

[87] Q.-Y. Zhou, J. Park, and V. Koltun, "Open3D: A Modern Library for 3D Data Processing," 1 2018. [Online]. Available: http://arxiv.org/abs/1801.09847 [Page 42.]

[88] "ICP registration - Open3D 0.19.0 documentation." [Online]. Available: https://www.open3d.org/docs/release/tutorial/pipelines/icp_registration.html [Pages 42 and 43.]

[89] “Colored point cloud registration - Open3D 0.19.0 documentation.” [Online]. Available: https://www.open3d.org/docs/release/tutorial/pipelines/colored_pointcloud_registration.html [Page 43.]

[90] “GitHub - DLR-RM/s3li-toolkit.” [Online]. Available: https://github.com/DLR-RM/s3li-toolkit [Pages 50 and 84.]

Appendix A

Supporting materials

A.1 Viewpoint Overlap Computation Functions

The following functions were implemented to evaluate the geometric and angular overlap between pairs of image frames in the Etna dataset. These were used to define ground truth loop closure matches based on spatial alignment and viewpoint consistency.

A.1.1 compute_overlap_v1

Listing A.1: Custom overlap scoring function based on angular difference and sigmoid-based corrections in the lateral and longitudinal directions.

```
def compute_overlap_v1(pos0, ang0, pos1, ang1,
    hor_fov = 60.0):
      ang0_positive = (180.0 * ang0 / np.pi) % 360
      ang1_positive = (180.0 * ang1 / np.pi) % 360
      ang_difference = 180 - np.abs(np.abs(
          ang0_positive - ang1_positive) - 180)

      angular_overlap_ratio = max(hor_fov - abs(
          ang_difference), 0.0) / hor_fov

      lateral_distance, longitudinal_distance =
          lateral_longitudinal_distances(pos0, ang0,
          pos1)
```

```
    position_correction_lateral = 1.0 - 1.0 / (1.0
       + np.exp(-lateral_distance + 80.0))
    position_correction_forward = 1.0 - 1.0 / (1.0
       + np.exp(-longitudinal_distance + 60.0))

    return angular_overlap_ratio * min(
       position_correction_lateral,
       position_correction_forward), \
            ang_difference, lateral_distance,
               longitudinal_distance
```

A.1.2 compute_overlap_v2

Listing A.2: Alternative method that computes the intersection area between the FOVs of two camera poses as polygons.

```
def compute_overlap_v2(pos0, ang0, pos1, ang1,
   hor_fov=45.0, fov_range1=75.0, fov_range2=75.0):
    fov0 = get_fov_triangle(pos0, ang0, hor_fov,
       fov_range1)
    fov1 = get_fov_triangle(pos1, ang1, hor_fov,
       fov_range2)

    poly0 = Polygon(fov0)
    poly1 = Polygon(fov1)
    intersection = poly0.intersection(poly1)
    intersection_area = intersection.area if
       intersection.is_valid else 0.0

    fov_area = max(Polygon(fov0).area, Polygon(fov1
       ).area)
    overlap_ratio = intersection_area / fov_area if
        fov_area != 0 else 0.0

    ang_difference = abs(ang0 - ang1) % 360
    ang_difference = 180 - abs(ang_difference -
       180)
    lateral_distance, longitudinal_distance =
       lateral_longitudinal_distances(pos0, ang0,
```

```
    pos1)

return overlap_ratio, ang_difference,
    lateral_distance, longitudinal_distance
```

TRITA – EECS-EX 2024:0000
Stockholm, Sweden 2025

www.kth.se

€€€€ For DIVA €€€€

{
”Author1”: { ”Last name”: ”Encinar Gonzalez”,
”First name”: ”Laura Alejandra”,
”Local User Id”: ”https://orcid.org/0009-0002-0509-1207”,
”E-mail”: ”laeg@kth.se”,
”organisation”: {”L1”: ”School of Electrical Engineering and Computer Science”,
}
},
”Cycle”: ”2”,
”Course code”: ”DA258X”,
”Credits”: ”30.0”,
”Degree1”: {”Educational program”: ”Master’s Programme, ICT Innovation, 120 credits”
,”programcode”: ”TIVNM”
,”Degree”: ”Master degree”
,”subjectArea”: ”Technology”
},
”Title”: {
”Main title”: ”Multi-Modal Place Recognition and Pose Estimation for Autonomous Rovers in Unstructured Environments”,
”Subtitle”: ”From Image Retrieval to 6D Pose Estimation for Loop Closure in SLAM”,
”Language”: ”eng” },
”Alternative title”: {
”Main title”: ”Multi-Modal Platsigenkänning och Positionsuppskattning för Autonoma Rovers i Ostrukturerade Miljöer”,
”Subtitle”: ”Från bildtagning till 6D lägesbestämning för loop-stängning i SLAM”,
”Language”: “swe”
},
”Supervisor1”: { ”Last name”: ”Folkesson”,
”First name”: ”John”,
”Local User Id”: ”https://orcid.org/0000-0002-7796-1438”,
”E-mail”: ”johnf@kth.se”,
”organisation”: {”L1”: ”School of Electrical Engineering and Computer Science”,
”L2”: ”Intelligent Systems” }
},
”Supervisor2”: { ”Last name”: ”Giubilato”,
”First name”: ”Riccardo”,
”E-mail”: ”riccardo.giubilato@dlr.de”,
”Other organisation”: ”German Aerospace Center (DLR), Institute of Robotics and Mechatronics”
},
”Supervisor3”: { ”Last name”: ”Zhou”,
”First name”: ”Quan”,
”E-mail”: ”quan.zhou@aalto.fi”,
”Other organisation”: ”Aalto University”
},
”Examiner1”: { ”Last name”: ”Jensfelt”,
”First name”: ”Patric”,
”Local User Id”: ”https://orcid.org/0000-0002-1170-7162”,
”E-mail”: ”patric@kth.se”,
”organisation”: {”L1”: ”School of Electrical Engineering and Computer Science”,
”L2”: ”Intelligent Systems” }
},
"National Subject Categories": "10201, 10206",
”Other information”: {”Year”: ”2025”, ”Number of pages”: ”xviii,109”},
”Copyrightleft”: ”copyright”,
”Series”: { ”Title of series”: ”TRITA – EECS-EX” , ”No. in series”: ”2024:0000” },
”Opponents”: { ”Name”: ”A. B. Normal & A. X. E. Normalè”},
”Presentation”: { ”Date”: ”2022-03-15 13:00”
,”Language”:”eng”
,”Room”: ”via Zoom https://kth-se.zoom.us/j/ddddddddddd”
,”Address”: ”Isafjordsgatan 22 (Kistagången 16)”
,”City”: ”Stockholm” },
”Number of lang instances”: ”3”,
”Abstract[eng]”: €€€€

Autonomous navigation in planetary-like environments presents unique challenges due to the absence of GPS signals, limited semantic structure, and visual ambiguity caused by repetitive textures or harsh lighting conditions. Traditional place recognition and localization methods either rely on dense maps and structured environments or only provide coarse retrieval without estimating full 6-DoF (Degrees of Freedom) poses. This limits their applicability in the context of real-time Simultaneous Localization and Mapping (SLAM) for field robotics and planetary exploration.

This thesis addresses the problem by developing a multi-modal system that performs both place recognition and relative pose estimation in unstructured, GNSS-denied environments. The proposed approach fuses visual features extracted from a transformer-based encoder (DINOv2) with 3D geometric descriptors from a LiDAR-based backbone (SONATA). These features are projected and aligned in 3D space to produce interpretable correspondences, from which the system estimates full 6D poses. On the retrieval side, DINOv2 descriptors are aggregated using SALAD, a learned VLAD-style module, and

searched efficiently using FAISS indexing. The system is evaluated on the Etna volcano dataset, representative of planetary terrains.

The results show that the proposed model outperforms established retrieval methods like NetVLAD and TransVPR and achieves more stable pose estimation than handcrafted or regression-based alternatives. The fusion of LiDAR and vision improved robustness in scenes with low texture or poor illumination, validating the hypothesis that multi-modality can bridge the gap between accuracy and generalization. Importantly, the system produces interpretable outputs and operates within real-time constraints for retrieval, although further optimization is needed for pose estimation.

This thesis demonstrates that it is feasible to move beyond retrieval-only frameworks and provide full, explainable 6D poses suitable for SLAM. Future work should focus on improving runtime efficiency in the pose estimation module, incorporating more diverse datasets, and testing deployment on real robotic platforms. These developments could contribute to more autonomous and trustworthy robotic systems for exploration, disaster response, and agriculture in extreme environments.

€€€€,
"Keywords[eng]": €€€€
Multi-modal place recognition, Six degrees of freedom pose estimation, Simultaneous Localization and Mapping (SLAM) integration, Transformer-based encoders, Light Detection and Ranging (LiDAR), DINO version 2, SONATA, Feature aggregation, Sinkhorn Algorithm for Locally Aggregated Descriptors (SALAD), Unstructured planetary environments, Real-time retrieval €€€€,
"Abstract[swe]": €€€€

Autonom navigering i planetliknande miljöer innebär unika utmaningar på grund av avsaknad av GPS-signaler, begränsad semantisk struktur och visuell tvetydighet orsakad av repetitiva texturer eller svåra ljusförhållanden. Traditionella metoder för platsigenkänning och lokalisering förlitar sig antingen på täta kartor och strukturerade miljöer eller erbjuder endast grov återhämtning utan att uppskatta fullständiga 6-\gls{DoF} (sex frihetsgrader) poser. Detta begränsar deras användbarhet i realtids-SLAM (Simultaneous Localization and Mapping) för fältrobotik och planetutforskning.

Denna avhandling angriper problemet genom att utveckla ett multimodalt system som utför både platsigenkänning och relativ posuppskattning i ostrukturerade miljöer utan GNSS. Den föreslagna metoden kombinerar visuella egenskaper extraherade från en transformerbaserad kodare (DINOv2) med 3D-geometriska beskrivare från en LiDAR-baserad ryggrad (SONATA). Dessa egenskaper projiceras och justeras i 3D-rymden för att generera tolkbara korrespondenser, från vilka systemet uppskattar fullständiga 6D-poser. På återhämtningssidan aggregeras DINOv2-beskrivare med hjälp av SALAD, en inlärd VLAD-liknande modul, och söks effektivt med FAISS-indexering. Systemet utvärderas på Etna-vulkanens datamängd, som är representativ för planetära terränger.

Resultaten visar att den föreslagna modellen överträffar etablerade metoder för återhämtning såsom NetVLAD och TransVPR, samt uppnår mer stabil posuppskattning än handgjorda eller regressionsbaserade alternativ. Kombinationen av LiDAR och visuella data förbättrade robustheten i scener med låg textur eller dålig belysning, vilket bekräftar hypotesen att multimodalitet kan överbrygga gapet mellan noggrannhet och generalisering. Viktigt är att systemet genererar tolkbara resultat och fungerar inom realtidskrav för återhämtning, även om vidare optimering krävs för posuppskattningen.

Denna avhandling visar att det är möjligt att gå bortom enbart återhämtningsbaserade ramverk och tillhandahålla fullständiga, förklarliga 6D-poser som lämpar sig för SLAM. Framtida arbete bör fokusera på att förbättra prestandan i posuppskattningsmodulen, inkludera mer varierade datamängder och testa implementering på verkliga robotplattformar. Dessa framsteg kan bidra till mer autonoma och tillförlitliga robotsystem för utforskning, katastrofinsatser och jordbruk i extrema miljöer.

€€€€,
"Keywords[swe]": €€€€
Multi-modal platsigenkänning, Sex frihetsgraders posuppskattning, Integration av simultan lokalisering och kartläggning (SLAM), Transformerbaserade kodare, Ljusdetektering och avståndsmätning (LiDAR), DINO version 2, SONATA, Funktionell aggregering, Sinkhorn-algoritm för lokalt aggregerade beskrivare (SALAD), Ostrukturerade planetära miljöer, Återhämtning i realtid €€€€,
"Abstract[fre]": €€€€

Autonominen navigointi planeettamaisissa ympäristöissä tuo mukanaan erityisiä haasteita GPS-signaalien puuttumisen, rajallisen semanttisen rakenteen sekä visuaalisen epäselvyyden vuoksi, jota aiheuttavat toistuvat tekstuurit ja vaikeat valaistusolosuhteet. Perinteiset paikan tunnistus- ja paikannusmenetelmät perustuvat joko tiheisiin karttoihin ja jäsenneltyihin ympäristöihin tai tarjoavat vain karkean haun ilman täysimääräistä 6-\gls{DoF} (kuuden vapausasteen) asennon estimointia. Tämä rajoittaa niiden soveltuvuutta reaaliaikaiseen SLAM-järjestelmään (Simultaneous Localization and Mapping) kenttärobotiikassa ja planeettojen tutkimuksessa.

Tämä diplomityö käsittelee ongelmaa kehittämällä multimodaalisen järjestelmän, joka suorittaa sekä paikan tunnistusta että suhteellisen asennon estimointia jäsentymättömissä, GNSS-vapaissa ympäristöissä. Ehdotettu lähestymistapa yhdistää transformer-pohjaisesta kooderista (DINOv2) poimitut visuaaliset piirteet LiDAR-pohjaiseen runkoon (SONATA) perustuvien 3D-geometristen piirteiden kanssa. Nämä piirteet projisoidaan ja kohdistetaan 3D-avaruudessa tuottaen tulkittavia vastaavuuksia, joiden perusteella järjestelmä arvioi täydet 6D-asennot. Haun osalta DINOv2-piirteet yhdistetään SALAD-menetelmällä, joka on oppiva VLAD-tyylinen moduuli, ja haku toteutetaan tehokkaasti FAISS-indeksoinnin avulla. Järjestelmä arvioitiin Etna-tulivuoren tietoaineistolla, joka edustaa planeettamaista maastoa.

Tulokset osoittavat, että ehdotettu malli päihittää vakiintuneet hakumenetelmät kuten NetVLAD ja TransVPR, ja saavuttaa vakaamman asennon estimoinnin kuin käsintehdyt tai regressiopohjaiset vaihtoehdot. LiDARin ja visuaalisen tiedon yhdistäminen paransi järjestelmän kestävyyttä alhaisen

tekstuurin tai heikon valaistuksen tilanteissa, vahvistaen hypoteesin siitä, että multimodaalisuus voi kuroa umpeen tarkkuuden ja yleistettävyyden välistä kuilua. Tärkeää on, että järjestelmä tuottaa tulkittavia tuloksia ja toimii reaaliaikaisissa hakuvaatimuksissa, vaikka asennon estimointimoduuli vaatii edelleen optimointia.

Tämä diplomityö osoittaa, että on mahdollista siirtyä pelkästään hakuun perustuvista järjestelmistä kohti täysiä, selitettävissä olevia 6D-asentoja, jotka soveltuvat SLAMiin. Tulevassa työssä tulisi keskittyä asennon estimoinnin suoritustehokkuuden parantamiseen, monipuolisempien tietoaineistojen käyttöönottoon sekä järjestelmän testaamiseen oikeilla robottialustoilla. Nämä kehitysaskeleet voivat edistää autonomisempien ja luotettavampien robottijärjestelmien kehitystä tutkimukseen, katastrofivalmiuteen ja maatalouteen äärimmäisissä olosuhteissa.

€€€€,
"Keywords[fre]": €€€€
Monimodaalinen paikantunnistus, Kuuden vapausasteen asentopositiomittaus, Samanaikainen paikannus ja kartoitus (SLAM) -integraatio, Transformer-pohjaiset kooderit, Valotutka (LiDAR), DINO versio 2, SONATA, Piirrekoosteet, Sinkhorn-algoritmi paikallisesti koottuja piirteitä varten (SALAD), Jäsentymättömät planetaariset ympäristöt, Reaaliaikainen haku €€€€,
}

acronyms.tex

```
%%% Local Variables:
%%% mode: latex
%%% TeX-master: t
%%% End:
% The following command is used with glossaries-extra
\setabbreviationstyle[acronym]{long-short}
% The form of the entries in this file is \newacronym{label}{acronym}{phrase}
%                                    or \newacronym[options]{label}{acronym}{phrase}
% see "User Manual for glossaries.sty" for the  details about the options, one example is shown below
% note the specification of the long form plural in the line below
\newacronym[longplural={Debugging Information Entities}]{DIE}{DIE}{Debugging Information Entity}
%
% The following example also uses options
\newacronym[shortplural={OSes}, firstplural={operating systems (OSes)}]{OS}{OS}{operating system}

% note the use of a non-breaking dash in long text for the following acronym
\newacronym{IQL}{IQL}{Independent -QLearning}

% example of putting in a trademark on first expansion
\newacronym[first={NVIDIA OpenSHMEM Library (NVSHMEM\texttrademark)}]{NVSHMEM}{NVSHMEM}{NVIDIA OpenSHMEM Library}

\newacronym{KTH}{KTH}{KTH Royal Institute of Technology}


% MY ACRONYMS
\newacronym{VPR}{VPR}{Visual Place Recognition}
\newacronym{ICT}{ICT}{Information and Communication Technology}
\newacronym{SLAM}{SLAM}{Simultaneous Localization and Mapping}
\newacronym{GPS}{GPS}{Global Positioning System}
\newacronym{GNSS}{GNSS}{Global Navigation Satellite System}
\newacronym{LiDAR}{LiDAR}{Light Detection and Ranging}
\newacronym{SIFT}{SIFT}{Scale-Invariant Feature Transform}
\newacronym{SURF}{SURF}{Speeded-Up Robust Features}
\newacronym{DoG}{DoG}{Difference-of-Gaussian}
\newacronym{ORB}{ORB}{Oriented FAST and Rotated BRIEF}
\newacronym{LoFTR}{LoFTR}{Local Feature TRansformer}
\newacronym{VLAD}{VLAD}{Vector of Locally Aggregated Descriptors}
\newacronym{CNN}{CNN}{Convolutional Neural Network}
\newacronym{GNN}{GNN}{Graph Neural Network}
\newacronym{ViT}{ViT}{Vision Transformer}
\newacronym{FPN}{FPN}{Feature Pyramid Network}
\newacronym{CLS}{CLS}{Classification Token}
\newacronym{MLP}{MLP}{Multi-layer Perceptrons}
\newacronym{GAP}{GAP}{Global Average Pooling}
\newacronym{GMP}{GMP}{Global Max Pooling}
\newacronym{GeM}{GeM}{Generalized Mean Pooling}
\newacronym{GGeM}{GGeM}{Group Generalized Mean Pooling}
\newacronym{NN}{NN}{Nearest Neighbors}
\newacronym{ANN}{ANN}{Approximate Nearest Neighbor}
\newacronym{FAISS}{FAISS}{Facebook AI Similarity Search}
\newacronym{IVF}{IVF}{inverted file}
\newacronym{PQ}{PQ}{product quantization}
\newacronym{HNSW}{HNSW}{Hierarchical Navigable Small World graphs}
\newacronym{BoW}{BoW}{Bag-of-Words}
\newacronym{LSH}{LSH}{Locality-Sensitive Hashing}
\newacronym{HLoc}{HLoc}{Hierarchical Localization}
\newacronym{VIO}{VIO}{Visual-Inertial Odometry}
\newacronym{IMU}{IMU}{Inertial Measurement Unit}
\newacronym{RPR}{RPR}{relative pose regression}
\newacronym{EMM}{EMM}{Essential Matrix Module}
\newacronym{DSAC}{DSAC}{Differentiable RANSAC}
\newacronym{PnP}{PnP}{Perspective-n-Point}
\newacronym{ICP}{ICP}{Iterative Closest Point}
\newacronym{SfM}{SfM}{Structure-from-Motion}
\newacronym{LRU}{LRU}{Lightweight Rover Unit}
\newacronym{UMF}{UMF}{Unifying Local and Global Multi-modal Features}
\newacronym{MAE}{MAE}{Masked Autoencoder}
\newacronym{FPFH}{FPFH}{Fast Point Feature Histograms}
\newacronym{BeV}{BeV}{'Birds-eye View}
\newacronym{FOV}{FOV}{Field of View}
\newacronym{PCA}{PCA}{Principal Component Analysis}
\newacronym{SDG}{SDG}{Sustainable Development Goals}
\newacronym{UN}{UN}{United Nations}
\newacronym{ECA}{ECA}{Efficient Channel Attention}
\newacronym{DoF}{DoF}{Degrees of Freedom}
\newacronym{BRIEF}{BRIEF}{Binary Robust Independent Elementary Features}
```

```
\newacronym{RANSAC}{RANSAC}{RANdom SAmple Consensus}
\newacronym{RGB}{RGB}{Red, Green, and Blue}
\newacronym{DLR}{DLR}{German Aerospace Center}
```