Universidad Zaragoza
1542

# Final Master Thesis

# Visual Relocalization from Sparse Views in Aliased and Low-Texture Environments via Novel View Synthesis

# Relocalización visual desde vistas dispersas en entornos con aliasing y baja textura mediante síntesis de nuevas vistas


Author

María Peribáñez Tafalla

Supervisors

Riccardo Giubilato

Javier Civera Sancho

ESCUELA DE INGENIERÍA Y ARQUITECTURA


2026

# ACKNOWLEDGMENTS

This work would not have been possible without the people who accompanied me throughout this journey.

First of all, I would like to thank my supervisors, Javier and Riccardo, for giving me the opportunity to work at an institution such as DLR. This experience has been incredibly enriching, both professionally and personally, and I am truly grateful for their trust and support throughout the development of this thesis.

I am also deeply thankful to all the colleagues I met during my time here. Beyond the shared knowledge, it was the everyday moments that made this experience truly special: the coffee breaks, the walks, the spontaneous conversations, and the laughter in the office. Those small moments turned work into something much more meaningful and made DLR feel like home. In particular, I would like to thank Leon, Matthias, and Yasin, my office mates, who became an essential part of my daily life, and Kristian, Kristina and Jakob from the office next door, whose company, and good energy made every day a little brighter.

I would also like to thank María, with whom I decided to embark on this adventure of living abroad. Sharing this experience with her has been an absolute privilege. Her support, companionship, and the countless moments we shared along the way made this journey even more special, and I could not have wished for a better person by my side.

Finally, I want to thank my family for always being there for me. Their constant support, encouragement, and unconditional belief in me gave me the confidence to step out of my comfort zone and embrace opportunities like this one. Everything I have achieved here would not have been possible without them.

To all of you, thank you for being part of this chapter of my life and for walking alongside me during this journey.

# ABSTRACT

Visual relocalisation is a fundamental component of robotic navigation systems, particularly in scenarios where global positioning systems are unavailable and the robot must rely exclusively on onboard perception to estimate its pose within a known environment. This problem is especially challenging in unstructured outdoor and planetary-like environments, characterized by low-texture surfaces, repetitive visual patterns, harsh illumination, and sparse, weakly overlapping viewpoints. Under these conditions, classical feature-based relocalisation pipelines suffer from severe performance degradation due to perceptual aliasing and limited geometric parallax.

This thesis addresses visual relocalisation in such challenging environments by leveraging Novel View Synthesis techniques, with a particular focus on explicit scene representations based on 3D Gaussian Splatting (3DGS). An initial analysis highlights the limitations of baseline Gaussian Splatting in planetary-like outdoor scenarios, revealing failure modes such as poor metric consistency, overfitting to photometric supervision, and unstable geometry in low-texture regions.

To overcome these limitations, a geometry-aware framework is proposed. Gaussian primitives are initialized using external point cloud priors; dense depth and surface normal supervision is incorporated using multi-view depth predictions; and global metric consistency is enforced through a LiDAR-guided geometric constraint formulated as a Chamfer-based loss. These complementary strategies are progressively combined to improve the structural fidelity and scale consistency of the reconstructed scenes while preserving photometric quality.

The proposed approach is evaluated on a real-world planetary dataset acquired by a mobile rover equipped with monocular RGB cameras and a solid-state LiDAR sensor. Results show substantial improvements in geometric accuracy over baseline Gaussian Splatting, achieving over a 90% reduction in geometric error while maintaining competitive photometric performance. Moreover, the improved scene representations lead to significant gains in downstream single-image 6-DoF camera relocalisation accuracy.

Overall, this work demonstrates that incorporating explicit geometric supervision into Gaussian Splatting representations is key to achieving reliable visual relocalisation in extreme outdoor environments, providing a robust and scalable solution for long-term autonomous robotic operation in planetary exploration scenarios.

# Index of Contents


**1 Introduction** **1**

1.1 Motivation . . . . . 1

1.2 Objectives and Contributions . . . . . 3

1.3 Context and tools . . . . . 4

1.4 Thesis structure . . . . . 5

**2 Related work** **7**

2.1 Principles of Novel View Synthesis and Gaussian Splatting . . . . . 7

2.1.1 Fundamentals of Novel View Synthesis . . . . . 7

2.1.2 Gaussian Splatting for Explicit Scene Representation . . . . . 8

2.1.3 Strengths and Limitations in Outdoor Robotics Scenarios . . . 9

2.2 Place recognition and Camera Pose Estimation . . . . . 11

2.2.1 Place Recognition . . . . . 11

2.2.2 Pose estimation . . . . . 13

**3 Methodology** **17**

3.1 Baseline 3D Gaussian Splatting . . . . . 17

3.1.1 Observed failure modes in planetary-like outdoor sequences . . 17

3.2 Initialization with Point Cloud Priors . . . . . 18

3.3 Depth-guided 3D Gaussian Splatting . . . . . 21

3.4 Geometric supervision for 3DGS . . . . . 22

3.4.1 Chamfer loss . . . . . 23

3.4.2 LiDAR supervision as a geometric prior . . . . . 23

3.4.3 Depth-aware weighting and training strategy . . . . . 24

3.5 Camera Pose Estimation from a 3DGS . . . . . 25

3.5.1 Place Recognition and Candidate Retrieval . . . . . 25

3.5.2 Single-image camera relocalisation . . . . . 26

**4 Evaluation** **29**

4.0.1 Dataset and sensors . . . . . 29

4.1 Experimental Evaluation of Geometry-aware Gaussian Splatting . . . . 31
4.1.1 Compared methods . . . . . 31
4.1.2 Training configurations . . . . . 32
4.1.3 Evaluation metrics . . . . . 32
4.1.4 Results . . . . . 34
4.2 Evaluation on Camera Pose Estimation . . . . . 38
4.2.1 Experimental setup . . . . . 38
4.2.2 Evaluation metrics . . . . . 39
4.2.3 Place recognition retrieval performance . . . . . 40
4.2.4 Camera Relocalisation Results . . . . . 41
4.3 Discussion . . . . . 43

**5 Conclusions** **47**
5.1 Summary . . . . . 47
5.2 Challenges and Limitations . . . . . 47
5.3 Future work . . . . . 48

**6 Bibliography** **49**

**Appendices** **54**

**A Dataset preprocessing for Place Recognition** **55**
A.1 Preprocessing pipeline . . . . . 55
A.2 Keyframe-based image selection . . . . . 56

# Chapter 1

# Introduction

## 1.1 Motivation

**Visual relocalisation** consists of estimating the camera pose within a global reference frame from its visual input and a global representation of the environment. It is a fundamental component of robotic navigation systems, particularly within SLAM pipelines, where it enables robust place recognition and loop closure [1, 2]. These capabilities are essential for maintaining globally consistent maps and accurate localisation.

Most existing relocalisation pipelines are designed for structured and texture-rich environments, where reliable keypoints and repeatable local descriptors can be extracted and matched across viewpoints. In these settings, relocalisation is commonly addressed through a combination of visual place recognition [3] to retrieve candidate locations and geometric verification to estimate a six degrees of freedom (6-DoF) camera pose. However, their performance degrades significantly when local features become unreliable or when viewpoint overlap is limited.

The problem of relocalisation becomes even more critical in planetary exploration scenarios, where GNSS is unavailable and robots must rely exclusively on onboard perception for drift correction and long-term autonomy.

**Limitations of visual relocalisation in planetary-like environments.** Figure 1.1 illustrates several challenges commonly found in planetary-like terrains. In this context, terrains suffer from low-texture surfaces, repetitive patterns, harsh illumination conditions, and large appearance changes, which lead to severe perceptual aliasing. In addition, image acquisition is often characterised by sparse and weakly overlapping viewpoints due to constrained rover trajectories. Images are usually captured along near-linear paths with little lateral displacement, resulting in limited baselines and weak geometric parallax. All of this limits the effectiveness of multi-view

geometry, making depth triangulation, scale estimation, and geometric verification unreliable. This makes classical feature-based relocalisation methods, which rely on local feature extraction and matching followed by geometric verification, prone to failure in such challenging environments.

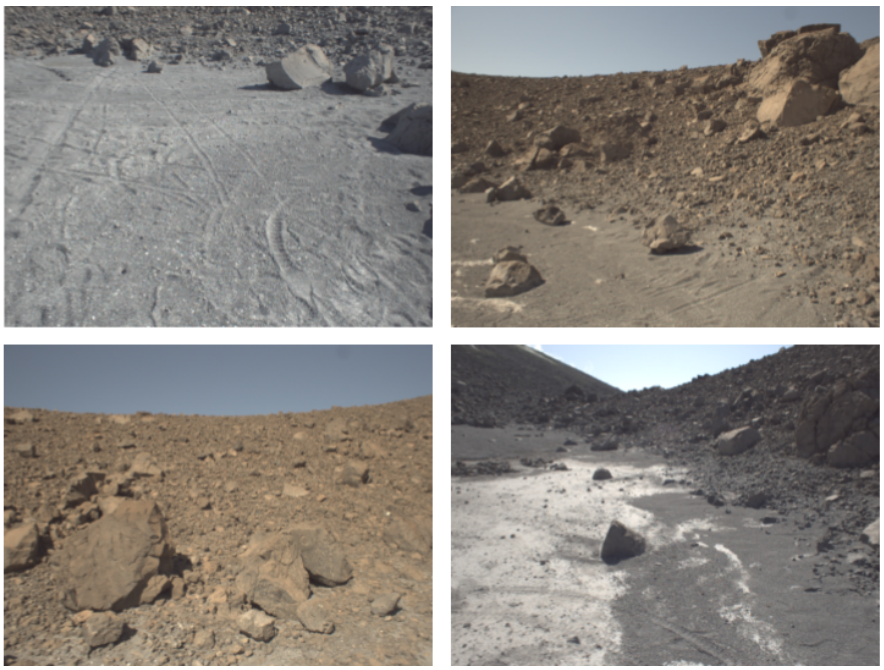


Figure 1.1: Example images illustrating typical challenges in planetary-like environments, including low-texture surfaces, perceptual aliasing, near-linear rover trajectories, and weak geometric parallax.

**Novel View Synthesis (NVS)** refers to a class of methods that aim to reconstruct a scene from a set of visual observations and render it from previously unseen viewpoints. In the context of visual relocalisation, NVS provides an alternative to purely correspondence pipelines by enabling the generation of intermediate views from sparse data, increasing effective viewpoint overlap and facilitating the establishment of reliable correspondences. This is particularly relevant in planetary-like environments, where rover trajectories and sensing constraints severely limit viewpoint diversity.

Among recent NVS approaches, Gaussian Splatting enables the construction of explicit, continuous 3D scene representations that can be efficiently rendered while preserving geometric consistency across viewpoints.

While recent works have explored Gaussian Splatting in robotic and localisation settings, they have focused on well-textured indoor environments. In contrast, this work does not treat novel view synthesis as an end goal, but rather as a means to obtain geometrically consistent scene representations that support robust visual relocalisation under extreme conditions. Such scenarios remain largely unexplored and continue to

pose significant challenges, as highlighted by recent planetary datasets [4].

## 1.2 Objectives and Contributions

The objective of this thesis is to address the problem of visual relocalisation in challenging outdoor environments. To overcome these limitations, we explore Novel View Synthesis techniques, with a particular focus on how Gaussian Splatting behave in such scenarios and how their quality and geometric consistency can be improved.

Gaussian Splatting is investigated as a flexible scene representation that fuses geometry and appearance from a limited set of observations and enables the synthesis of intermediate viewpoints with increased visual overlap. A central objective of this thesis is to analyse the limitations of these representations in planetary-like environments and to introduce geometry-aware improvements that maximise their structural accuracy and metric consistency.

By enhancing the geometric fidelity of the reconstructed scenes, the resulting representations can provide more reliable support for correspondence establishment, enabling more accurate and robust 6-DoF camera pose estimation. In order to achieve the planned goals, the following tasks were executed throughout the project (its distribution along the months may be observed in Table 1.1):

- An analysis of the limitations of baseline Gaussian Splatting representations in planetary-like outdoor environments.
- The design and integration of geometry-aware improvements for Gaussian Splatting, including point cloud initialization, depth-guided supervision, and LiDAR-based geometric constraints, aimed at improving structural accuracy and metric consistency.
- An experimental evaluation of geometry-aware Gaussian Splatting reconstructions on real rover datasets, analysing their photometric quality, geometric fidelity, and robustness under challenging outdoor conditions.
- The application of the improved Gaussian Splatting representations to visual relocalisation, using the 6DGS (6D Pose Estimation from a Single Image and a 3D Gaussian Splatting Model) framework as an evaluation benchmark [5].
- An analysis of the impact of reconstruction geometry quality on 6-DoF camera pose estimation accuracy and robustness.

| Task | Aug | Sep | Oct | Nov | Dec | Jan |
|---|---|---|---|---|---|---|
| Literature review | | | | | | |
| Dataset preparation | | | | | | |
| Baseline 3DGS + initialization priors | | | | | | |
| Geometry-aware GS | | | | | | |
| Pose estimation pipeline | | | | | | |
| Evaluation | | | | | | |
| Writing | | | | | | |

Table 1.1: Gantt diagram representing the distribution of the tasks along the months.

## 1.3 Context and tools

This work is situated in the context of visual relocalisation for robotic navigation and investigates novel view synthesis techniques, with a particular focus on Gaussian Splatting, as a step towards robust localisation under extreme viewpoint and appearance changes.

Gaussian Splatting is implemented using the Nerfstudio framework, which provides a flexible and extensible environment for training and evaluation. Geometry-aware extensions are integrated into this framework by incorporating depth predictions and LiDAR-based geometric supervision.

The SLAM system used to generate camera poses, submaps, and ground-truth reference data is implemented in C++ and operates within the Robot Operating System (ROS) framework. This system provides realistic sensor data acquisition and consistent camera geometry.

All learning-based components are implemented in Python and rely on scientific computing and deep learning libraries, including NumPy and PyTorch.

Experiments are conducted on a local workstation equipped with a NVIDIA GeForce RTX 4090 GPU, enabling efficient training and evaluation of Gaussian Splatting models.[1]

During the development of this thesis, artificial intelligence tools were used to aid in data visualization, plot creation, and the automated processing of simulation results. They were also leveraged for code debugging, proofreading, and for assisting with the proper use of LaTeX commands, including table and figure formatting.

[1] All the code developed for this thesis is accessible upon request.

## 1.4 Thesis structure

The main contributions of this work are structured as follows:

- **Chapter 2** reviews the relevant literature on Novel View Synthesis and Gaussian Splatting. It discusses classical and learning-based approaches, highlighting their strengths and limitations under sparse viewpoints, low-texture conditions, and perceptual aliasing.
- **Chapter 3** presents the proposed methodology. It analyses the limitations of baseline 3D Gaussian Splatting in planetary-like environments and introduces a set of geometry-aware extensions, including point cloud initialization, depth-guided supervision using MVSAnywhere, and a LiDAR-guided Chamfer-based geometric prior to enforce metric scale consistency and structural fidelity. The chapter also describes the camera relocalisation pipeline built on top of the reconstructed Gaussian scenes.
- **Chapter 4** provides an extensive experimental evaluation of the proposed methods on real-world rover datasets. It assesses both photometric reconstruction quality and geometric accuracy, and analyses the impact of geometry-aware supervision on downstream camera pose estimation performance.
- Finally, **Chapter 5** summarises the main conclusions of the thesis and discusses potential directions for future work.

# Chapter 2

# Related work

## 2.1 Principles of Novel View Synthesis and Gaussian Splatting

This chapter introduces the fundamental principles of Novel View Synthesis (NVS), tracing the evolution from classical geometry-based methods to modern neural representations. We focus on 3D Gaussian Splatting (3DGS), analyzing its strengths and limitations when applied to planetary-like environments.

### 2.1.1 Fundamentals of Novel View Synthesis

Novel View Synthesis (NVS) addresses the challenge of generating photorealistic renderings from arbitrary viewpoints given a sparse set of input images [6]. In robotic perception, NVS is particularly relevant in scenarios with sparse viewpoints, where direct feature matching and multi-view geometry become unreliable.

Classical NVS approaches relied on explicit geometric reconstruction, through feature correspondences, triangulation, and view interpolation [7, 8]. While effective under controlled conditions, these methods typically require accurate camera calibration, sufficient visual texture, and dense viewpoint coverage, assumptions that are often lacking in real-world outdoor robotic scenarios.

In recent years, deep neural networks have enabled a step towards learning-based scene representations. Rather than reconstructing discrete geometry, such as point clouds or meshes, neural methods encode scenes as continuous functions learned directly from image observations. A landmark work in this direction is Neural Radiance Fields (NeRF) [9], which represents a scene as a continuous 5D function, mapping 3D spatial coordinates and 2D viewing direction to density and color at that spatial location. This function is parameterized by a multilayer perceptron (MLP) and optimized using differentiable volume rendering, allowing the synthesis of photorealistic images from any virtual viewpoints.

NeRF has achieved state-of-the-art visual quality and has inspired subsequent work. However, its reliance on volumetric rendering requires sampling a large number of points along each camera ray, making both training and inference computationally expensive and memory intensive. As a result, NeRF-based methods are often unsuitable for real-time or resource-constrained robotic applications. Moreover, the geometry in NeRF is learned implicitly through the density field, which can lead to noisy, incomplete, or poorly localized surface representations; an important limitation when accurate geometry is required for downstream tasks such as localisation or mapping.

### 2.1.2 Gaussian Splatting for Explicit Scene Representation

To address the computational and geometric limitations of volumetric neural rendering, 3D Gaussian Splatting (3DGS) [10] was introduced as an efficient point-based 3D reconstruction for real-time novel view synthesis. Unlike NeRF, which encodes scenes implicitly, 3DGS represents a scene explicitly as a collection of differentiable 3D Gaussian primitives.

In the standard 3DGS pipeline, the scene is initialized from a sparse point cloud obtained via Structure-from-Motion (SfM) or from depth maps. Although less common, random initialization of Gaussian primitives in 3D space has also been investigated as an alternative when geometric priors are unavailable.

Each point is modeled as an anisotropic 3D Gaussian characterized by its position, covariance, opacity, and color. The covariance is further decomposed into a scaling vector and a rotation quaternion, allowing each Gaussian to adapt its orientation and shape to the underlying scene structure.

During training, all the parameters of the Gaussians are optimized through gradient descent by minimizing a photometric reconstruction loss between rendered images and ground-truth training views. The loss function combines L1 pixel-wise error with D-SSIM (structural dissimilarity), a perceptual metric that emphasizes structural consistency over exact color matching:

$$L = (1 - \lambda)L_1 + \lambda L_{D\text{-}SSIM} \tag{2.1}$$

where $\lambda$ is a weighting coefficient typically set to 0.2 in the original 3DGS. While effective for textured scenes with dense views, this purely photometric objective does not penalize geometric inconsistencies, a limitation that becomes critical under sparse viewpoints.

In the process of Gaussian optimization, a key component of 3DGS is adaptive

densification for detailed reconstruction. As shown in Figure 2.1, the adaptive densification mechanism dynamically refines the Gaussian representation by cloning, splitting, and pruning primitives based on reconstruction quality during optimization. This strategy focuses on areas with missing geometric features or regions where Gaussians are over-expanded, both exhibiting large view-space positional gradients. For under-reconstructed areas, small Gaussians are cloned and moved towards the positional gradient direction to improve coverage. In over-reconstructed regions, large Gaussians with high variance are split into two smaller ones. Additionally, Gaussians that are transparent, with opacity less than a specific threshold, are pruned to reduce redundancy. This process leads to a progressively denser and more accurate point-based representation without sampling empty space, unlike ray-based volumetric methods.

Thanks to its explicit representation and efficient rasterization-based rendering, 3DGS enables real-time Novel View Synthesis while maintaining high visual fidelity, making it attractive for practical applications.

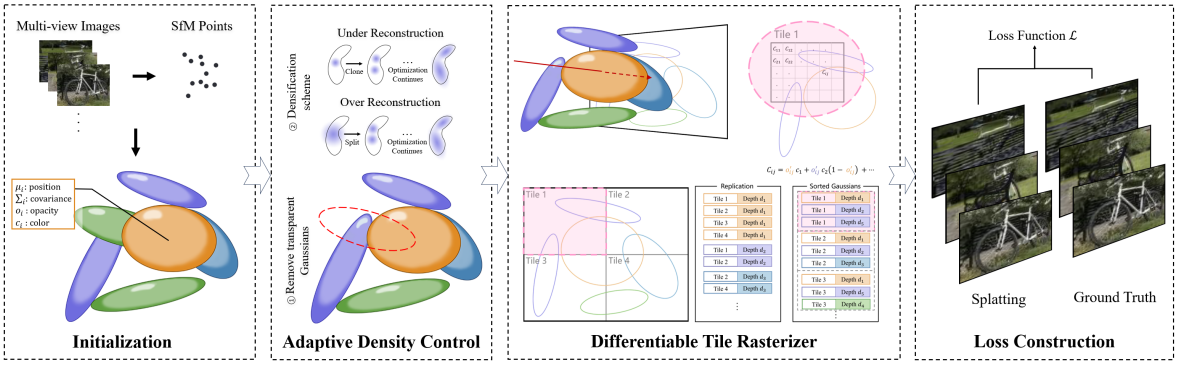



Figure 2.1: 3D Gaussian Splatting pipeline [11]. Sparse geometry from SfM initializes 3D Gaussians, which are optimized via photometric loss and adaptive densification (cloning, splitting, pruning) while maintaining real-time rendering through tile-based rasterization.

### 2.1.3 Strengths and Limitations in Outdoor Robotics Scenarios

Despite its advantages, 3D Gaussian Splatting faces significant challenges when working in real-world outdoor robotic environments [11]. One of the primary limitations arises in weakly textured regions, where traditional SfM pipelines struggle to generate reliable initial point clouds. Since 3DGS relies on such initialization, poor geometric seeds can propagate errors throughout the optimization process, leading to distorted or incomplete reconstructions.

In addition, without absolute depth or scale supervision, monocular Gaussian Splatting reconstructions suffer from metric ambiguity. While relative geometry

may be visually consistent, the resulting scene often exhibits scale drift, making distances and object sizes unreliable. In large outdoor environments, this can translate into substantial errors in estimated camera trajectories and scene geometry, severely limiting the usability of the reconstruction for downstream robotic tasks such as localisation or navigation.

Another critical issue in sparse-view settings is overfitting to the available training images. With limited overlapping viewpoints, Gaussians tend to optimise for photometric consistency with the observed views rather than for physically meaningful geometry. As a result, the representation may explain the training images correctly, while failing to generalise to novel viewpoints.

Recent research has explored various extensions to improve robustness under sparse-view and low-texture conditions. Some approaches focus on improving geometric supervision during training, while others aim to stabilize optimization or to integrate Gaussian representations into full SLAM systems [12]. For example, methods such as D2GS [13] and DepthSplat [14] incorporate monocular depth priors into the Gaussian Splatting framework. While these approaches improve geometric stability, they often rely on depth models pretrained on indoor or urban datasets, which can limit their generalisation to planetary-like outdoor terrains.

Beyond reconstruction, Gaussian representations have also been integrated into full SLAM systems. Gaussian-SLAM [15] is an example of this trend by adopting 3D Gaussians as the main scene representation within a real-time visual SLAM pipeline. Although such systems demonstrate improved robustness in dynamic environments, they typically suffer from significant computing expenses, making scalability to large outdoor scenes challenging.

Nevertheless, important research challenges remain. Balancing real-time performance with geometric accuracy, ensuring scale consistency across large outdoor scenes, and maintaining robustness under extreme viewpoint sparsity are still open problems. Furthermore, most evaluations of novel view synthesis rely on static image quality metrics (PSNR, SSIM, or LPIPS), which measure photometric fidelity but not geometric correctness.

These limitations motivate the integration of additional geometric priors into Gaussian Splatting. In the next chapter, we build upon this analysis by incorporating dense depth estimation and LiDAR-guided geometric supervision, with the aim of improving scale consistency and structural fidelity in challenging real-world rover datasets. In this thesis, Gaussian Splatting is therefore not treated as an end goal for novel view synthesis, but as a geometry-aware scene representation designed to support accurate and robust camera pose estimation in outdoor robotic environments.

Having established that 3DGS can provide explicit geometric scene representations despite its limitations in sparse outdoor settings, we now examine how such representations can be leveraged for camera relocalisation. The quality of the reconstructed geometry directly impacts the accuracy of pose estimation, motivating our integrated approach that addresses both reconstruction and localisation jointly.

## 2.2 Place recognition and Camera Pose Estimation

Place recognition and pose estimation play a critical role in ensuring the consistency of the SLAM generated map. In large-scale environments, the camera relocalisation approach is typically formulated as a two-stage problem: place recognition, which allows the robot to recognize previously visited locations, even if they are view from different angles or under different conditions; and camera pose estimation, which refines the camera's 6-DoF pose relating it with the previously recorded scene. These components are essential for loop closure detection, where the robot can identify if it has returned to a previously mapped area and can correct localisation drift. While this paradigm has proven effective in structured environments, it remains challenging in outdoor robotic scenarios.

To contextualize the approach adopted in this thesis, this section reviews traditional feature-based methods, learning-based place recognition techniques, recent advances in pose estimation from scene representations, and the challenges of integrating these components in sparse-view outdoor environments.

### 2.2.1 Place Recognition

Visual Place Recognition (VPR) aims to identify previously visited locations based on visual observations, typically under significant changes in viewpoint, illumination, and environmental conditions. The first and most critical stage of VPR is feature extraction, which consists of computing meaningful visual representations from raw images using either handcrafted or learned descriptors. These features are expected to be robust to viewpoint changes, lighting variations, and limited texture, while remaining discriminative across different locations.

#### Traditional feature-based approaches

Classical feature-based approaches rely on local keypoint detectors and descriptors such as SIFT [16], ORB [17], or learned alternatives like SuperPoint [18]. In structured and texture-rich environments, these methods enable reliable image matching and geometric verification. However, their performance degrades significantly in outdoor

robotic scenarios. In such conditions, reliable local correspondences become difficult to establish, leading to frequent relocalisation failures. These limitations are particularly pronounced in planetary-like terrains, where visual cues are sparse and perceptual aliasing is common.

As a result, purely local feature-based VPR approaches struggle to scale robustly to large, unstructured outdoor environments, motivating the use of higher-level representations that capture more global contextual information.

**Learning-based place recognition**

To overcome the limitations of local feature matching, learning-based place recognition methods based on global image descriptors have been widely explored. Early approaches relied on Bag-of-Words representations [19], while more recent methods employ convolutional neural networks to learn compact and discriminative global embeddings. A representative example is NetVLAD [20], which extends the classical VLAD formulation by introducing a differentiable pooling layer that aggregates local Convolutional Neural Networks (CNN) features into a single global descriptor. By learning both the feature extraction and the aggregation process end-to-end, NetVLAD enables efficient large-scale image retrieval and has become a common baseline in modern visual localisation pipelines.

Building upon this paradigm, several state-of-the-art methods have proposed improved aggregation strategies and training objectives to enhance robustness under viewpoint changes and perceptual aliasing. Approaches such as CosPlace [21] and MixVPR [22] leverage metric learning objectives to explicitly encourage viewpoint-invariant global descriptors, while methods like AnyLoc [23] aim to improve cross-domain generalisation by reducing dependence on dataset-specific biases. Although highly effective on standard VPR benchmarks, these CNN-based approaches often require large labeled datasets and careful domain adaptation, which can limit their applicability in environments where annotated data is scarce or unavailable.

Recent advances in self-supervised learning and transformer-based architectures have opened new directions for place recognition. Models such as DINO [24] learn general-purpose visual representations without requiring labeled data, capturing both semantic and structural information at multiple spatial resolutions. Unlike task-specific CNN descriptors, these representations exhibit strong transferability across domains and have shown promising performance when used as global image embeddings for place recognition in challenging and previously unseen environments.

In practice, global descriptor-based retrieval is typically used to identify a small set of candidate locations for a given query image, which are subsequently refined

through local geometric verification or pose estimation [25]. While this strategy improves robustness under large viewpoint and appearance changes, it does not directly provide metric pose information and remains sensitive to perceptual aliasing in environments with repetitive or self-similar structures. Consequently, accurate camera relocalisation requires coupling learning-based retrieval with a geometrically consistent scene representation capable of supporting reliable pose refinement.

### 2.2.2 Pose estimation

Pose estimation refers to the process of determining the position and orientation of a camera 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.

#### 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 [26], 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 [27], which iteratively selects minimal subsets of correspondences to identify a geometrically consistent solution. This PnP-RANSAC combination remains a reliable baseline in structure-based localisation tasks and is a core component in Structure-from-Motion pipelines such as COLMAP [28]. 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 [29]. 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.

### Learning-based pose estimation

More recently, learning-based pose regression approaches such as PoseNet [30] have been proposed to estimate camera pose from images using convolutional neural networks. While attractive due to their simplicity and real-time performance, these methods typically struggle with generalisation to new environments and lack explicit geometric consistency guarantees. Furthermore, their black-box nature makes them difficult to interpret and integrate into geometry-based SLAM pipelines, limiting their adoption in safety-critical robotic applications.

These limitations motivate approaches that leverage dense scene representations with explicit geometry, bridging place recognition and pose estimation within a unified framework.

### Pose estimation from scene representations

Beyond classical sparse maps, recent work has explored dense and continuous scene representations for camera pose estimation. Neural scene representations encode rich geometric and photometric information that can be directly exploited for pose refinement, enabling alternatives to correspondence-based localisation pipelines.

In this paradigm, camera pose estimation is formulated as a *render-and-compare* optimization problem: given an initial pose estimate, camera parameters are iteratively updated to minimize a measure between a rendered view of the scene and the observed query image. This formulation enables direct alignment between the image and the scene representation without requiring explicit feature matching. iNeRF [31] demonstrated this approach using Neural Radiance Fields, optimizing camera pose through gradient descent on photometric residuals. However, the high computational cost and implicit geometry of NeRF limit its applicability in real-time robotic scenarios.

More recently, Gaussian Splatting-based representations have emerged as efficient alternatives for pose estimation. By representing scenes as collections of explicit, differentiable 3D Gaussian primitives, 3DGS enables fast rasterization-based rendering and stable gradients with respect to camera parameters. This has led to several approaches that directly estimate camera pose by optimizing render-and-compare objectives on top of pretrained Gaussian Splatting models. In particular, the 6DGS framework formulates camera relocalisation as a differentiable optimization over a fixed 3D Gaussian scene, achieving accurate and efficient 6-DoF pose estimation.

Related efforts have also integrated Gaussian-based scene representations into full SLAM systems, such as Gaussian-SLAM [15], further demonstrating the suitability of Gaussian primitives for joint mapping and pose optimization. Across these approaches,

a common observation is that pose estimation accuracy is tightly coupled to the geometric fidelity and metric consistency of the underlying Gaussian representation. Inaccurate geometry or scale inconsistencies can lead to convergence to incorrect local minima, even when photometric alignment appears successful.

This strong dependence on scene geometry motivates the geometry-aware Gaussian Splatting framework introduced in Chapter 3, which aims to improve downstream camera relocalisation by enforcing geometric consistency during reconstruction.

# Chapter 3

# Methodology

This chapter presents a geometry-aware methodology to improve baseline 3DGS in planetary-like outdoor environments. Under these conditions, standard 3DGS often fails to produce metrically reliable scene reconstructions.

To address these limitations, we introduce a set of complementary strategies aimed at enhancing structural fidelity and scale consistency. First, we analyze the baseline 3DGS when relying solely on photometric supervision (Section 3.1). We then investigate geometric initialization using external priors (Section 3.2), introduce dense depth supervision via MVSAnywhere (Section 3.3), and enforce metric scale consistency through LiDAR-guided Chamfer loss (Section 3.4). Finally, we present our camera relocalisation approach leveraging these geometry-aware reconstructions (Section 3.5).

Together, these components establish a robust framework for scene reconstruction and pose estimation in challenging planetary-like environments. An overview of the complete methodology and its main components is shown in Figure 3.1.

## 3.1 Baseline 3D Gaussian Splatting

Baseline 3DGS [10] provides an efficient and high-quality framework for novel view synthesis by combining explicit geometry with learned appearance under photometric supervision. This baseline optimizes Gaussian parameters using photometric supervision and adaptive densification, and serves as the reference point upon which all geometry-aware extensions in this chapter are built.

### 3.1.1 Observed failure modes in planetary-like outdoor sequences

When applied to our planetary-like outdoor rover sequences, the baseline often produces renderings that are visually plausible but geometrically unreliable. In particular, we consistently observe: (i) blurred or over-smoothed surfaces, (ii) loss of sharp geometric

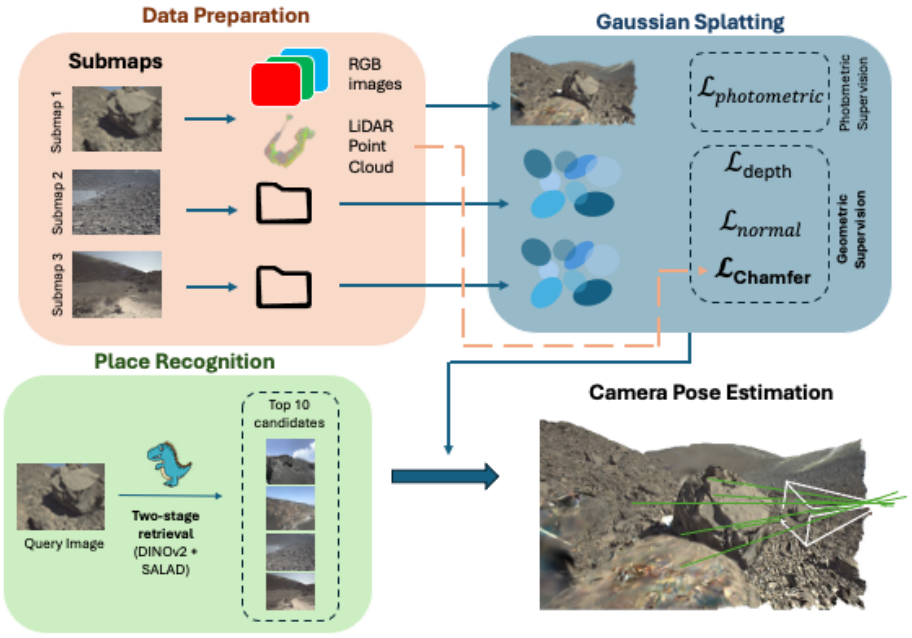



Figure 3.1: Overview of the proposed geometry-aware pipeline. RGB images and LiDAR point clouds are grouped into local submaps and used to train 3D Gaussian Splatting (3DGS) models under photometric and geometric supervision. LiDAR measurements provide explicit geometric constraints through a Chamfer loss, enforcing metric scale and structural consistency. At inference time, a single query image retrieves top-k candidate submaps using a two-stage place recognition pipeline based on DINOv2 and SALAD, and the camera pose is estimated by optimizing the query against the corresponding 3D Gaussian scene representation.

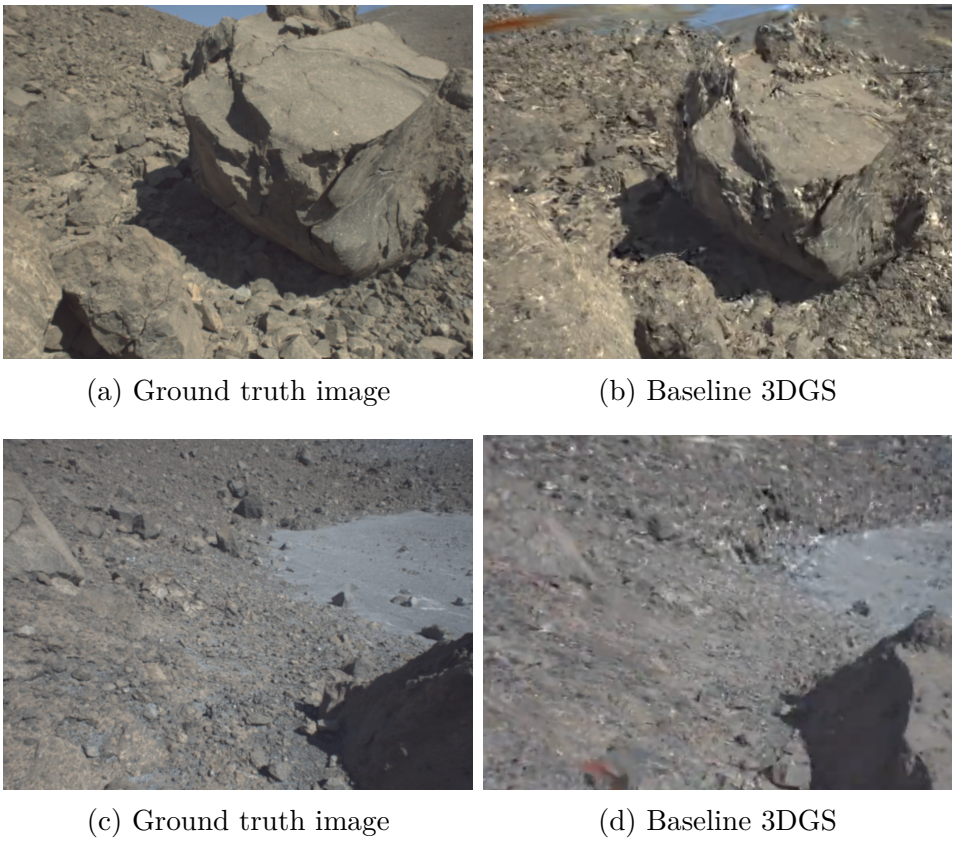
edges (e.g., rock boundaries), and (iii) floating artifacts and locally inconsistent structures in weakly textured regions.

Figure 3.2 illustrates these effects: although the synthesized views match the global appearance of the scene, fine-scale geometry is not preserved and surfaces lack crisp discontinuities. Overall, these limitations highlight the shortcomings of relying solely on photometric supervision in planetary-like environments and motivate the geometry-aware extensions introduced in the following sections.

## 3.2 Initialization with Point Cloud Priors

3DGS exhibits a strong dependency on its initial geometric configuration, as the optimization process refines the scene representation primarily through photometric supervision. Consequently, the spatial distribution and scale of the initial Gaussian primitives play an important role in the stability, convergence, and geometric consistency of the reconstruction.

In the absence of external geometric priors, 3DGS initializes the Gaussian primitives



(a) Ground truth image (b) Baseline 3DGS

(c) Ground truth image (d) Baseline 3DGS

Figure 3.2: Qualitative comparison between ground truth images and renderings produced by baseline 3DGS in a planetary-like outdoor environment.

with a random spatial distribution within a bounded scene volume. This default initialization does not constrain the metric scale of the scene nor enforce surface consistency, resulting in limited geometric guidance prior to photometric optimization.

A natural extension consists of initializing the Gaussian primitives using external point cloud priors. In this work, we consider point clouds obtained from different sources, including classical Structure-from-Motion (SfM) pipelines and LiDAR-based reconstructions. These priors provide complementary geometric information: SfM point clouds are visually consistent with the image observations but may be sparse or incomplete, whereas LiDAR measurements provide metrically accurate but sparse depth information.

### Complementary Nature of SfM and LiDAR Priors

SfM-based point clouds typically offer denser coverage in well-textured regions and are photometrically consistent with the training images, allowing them to capture fine surface details where reliable feature correspondences exist. However, SfM reconstructions often fail in low-texture or repetitive areas, suffer from scale ambiguity in the absence of absolute depth information, and are highly sensitive to sparse or

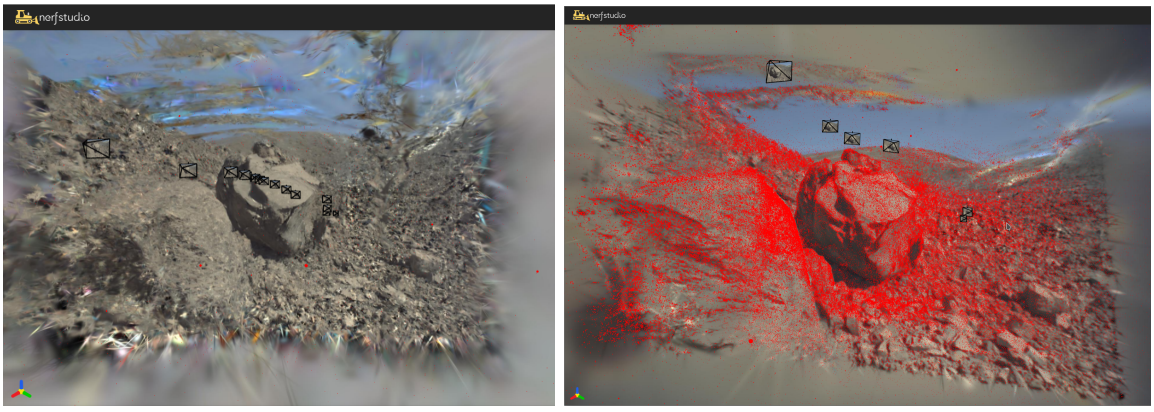
weakly overlapping viewpoints.

In contrast, LiDAR measurements provide metrically accurate absolute depth information and remain reliable even in textureless regions. Although LiDAR point clouds are typically sparse and may miss fine geometric details, they offer strong global scale constraints that are critical for outdoor robotic applications. Practical limitations such as occlusions, limited field of view, or reflective surfaces may further reduce their spatial coverage.

While initialization using either SfM or LiDAR priors provides a more structured starting point than random initialization, such priors remain passive: the subsequent optimization can still diverge or overfit to photometric supervision when geometric constraints are weak. This observation motivates the incorporation of active geometric supervision during training, which is introduced in Sections 3.3 and 3.4.

Figure 3.3 illustrates the default initialization in the Nerfstudio framework compared to the use of a point cloud prior. When working with initialization priors, the external point clouds are used solely to define the initial positions and scales of the Gaussian primitives.

The influence of quality of an initialization prior, stability, and metric consistency of the reconstructed scene is evaluated in the experimental section. While improved initialization can provide a more structured starting point, it may still be insufficient in challenging planetary environments, motivating the depth-guided and LiDAR-supervised training strategies introduced in the following sections.



(a) Random initialization (b) Prior inizialization

Figure 3.3: Comparison between the default random initialization of Gaussian primitives in the Nerfstudio framework and initialization using an external point cloud prior derived from SfM or LiDAR data. The red points indicate the initial positions of the Gaussian primitives before training. While prior-based initialization yields a more structured and geometrically meaningful starting configuration, it acts only as a passive constraint, as the subsequent optimization remains purely photometric.

## 3.3 Depth-guided 3D Gaussian Splatting

To reduce the geometric ambiguities observed in baseline 3DGS, this section introduces depth- and normal-guided training using the MVSAnywhere framework [32].

MVSAnywhere is a multi-view stereo architecture designed to generalize across diverse domains and depth ranges, enabling accurate depth estimation from multiple views in a wide variety of scene types. Unlike traditional multi-view stereo methods, it demonstrates strong generalization capabilities in challenging outdoor environments, making it particularly suitable for planetary-like terrains.

In this work, MVSAnywhere is employed as a geometric regularizer for 3DGS, following recent approaches such as VCR-GauS [33] and DN-Splatter [34]. The depth predictions produced by MVSAnywhere are incorporated into the training process as additional signals. The predicted depth maps are projected into 3D space using known camera intrinsics and poses, and are used to supervise the Gaussian-rendered depth and surface normals during optimization.

Figure 3.4 illustrates the effect of depth and normal-based regularization in 3DGS, as reported in the original MVSAnywhere work. Compared to the baseline, the use of depth and surface normal constraints leads to significantly more coherent and stable geometry.

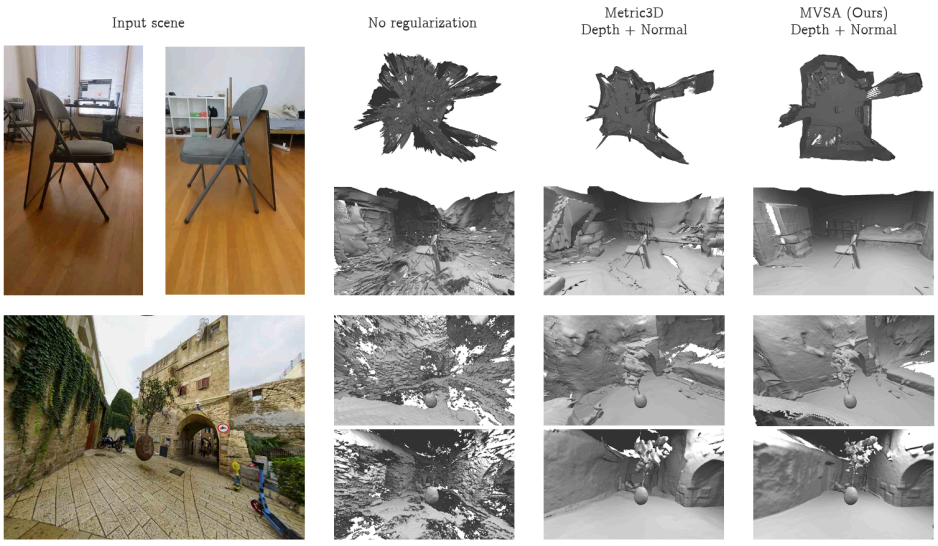



Figure 3.4: Qualitative comparison of 3DGS reconstructions with and without geometric regularization, using depth and surface normals predicted by MVSAnywhere. Image reproduced from the MVSAnywhere paper [32].

To guide the optimization of the Gaussian parameters, depth and surface normal consistency losses are introduced.

A direct L1 loss is applied over valid pixels:

$$\mathcal{L}_{\text{depth}}^{\text{metric}} = \frac{1}{|\Omega|} \sum_{i \in \Omega} \left| D_i^{\text{GS}} - D_i^{\text{MVS}} \right|,$$

where $D^{\text{GS}}$ denotes the depth rendered from the 3DGS model, $D^{\text{MVS}}$ the depth predicted by MVSAnywhere, and $\Omega$ the set of valid pixels.

Additionally, surface normal consistency is encouraged through a cosine similarity loss:

$$\mathcal{L}_{\text{normal}} = 1 - \frac{1}{|\Omega|} \sum_{i \in \Omega} \mathbf{n}_i^{\text{GS}} \cdot \mathbf{n}_i^{\text{MVS}}.$$

Surface normals are obtained by differentiating the predicted depth maps produced by MVSAnywhere.

Together, these losses improve local surface coherence and reduce depth ambiguities in poorly textured regions. Despite its strong generalization capabilities, MVSAnywhere still relies on geometric parallax between multiple views to accurately triangulate depth. In planetary-like outdoor scenarios, image acquisition is often characterized by sparse viewpoints and constrained rover trajectories, which limit the available baseline and reduce effective parallax. Under these conditions, depth estimation becomes poor, particularly for distant or weakly observed regions of the scene.

Although MVSAnywhere incorporates learned monocular cues that improve robustness, these priors remain insufficient to fully resolve global scale ambiguity or guarantee metric consistency across the reconstructed scene. As a result, depth-guided supervision alone cannot ensure reliable scale alignment, motivating the introduction of an explicit LiDAR-based geometric prior, which is presented in the following section.

## 3.4 Geometric supervision for 3DGS

Recent works have highlighted the importance of incorporating LiDAR-based geometric priors into 3DGS representations to improve metric consistency and structural accuracy in outdoor scenarios. Approaches such as GS-LiDAR [35] and SplatAD [36] demonstrate that explicitly aligning Gaussian-based scene representations with LiDAR measurements leads to improved geometric realism. Inspired by these observations, we introduce a LiDAR-guided Chamfer-based loss that enforces global scale consistency and geometric alignment during training.

While depth-guided supervision improves local geometric coherence, the reconstructed scene remains ambiguous up to an unknown global scale factor. This limitation is critical, as scale is essential for subsequent robotic tasks, such as

localisation or navigation. To address this issue, a new loss is introduced as an additional geometric restriction during training.

### 3.4.1 Chamfer loss

Geometric alignment between the predicted and reference point clouds is enforced through a symmetric formulation of the Chamfer distance. Chamfer distance is a commonly used metric to measure the similarity between two point sets without requiring explicit point-to-point correspondences. Given two point clouds, A and B, the metric is defined as the average distance from each point in one set to its nearest neighbor in the other set, computed in both directions.

Due to its simplicity and robustness, it is often used as a loss function for geometric alignment, encouraging the generated point cloud to be as close as possible to the target point cloud. In this context, the Chamfer loss is defined as:

$$\mathcal{L}_{\text{Chamfer}} = \frac{1}{|\mathcal{P}_{\text{GS}}|} \sum_{\mathbf{p} \in \mathcal{P}_{\text{GS}}} \min_{\mathbf{q} \in \mathcal{P}_{\text{LiDAR}}} \|\mathbf{p} - \mathbf{q}\|_2 + \lambda \frac{1}{|\mathcal{P}_{\text{LiDAR}}|} \sum_{\mathbf{q} \in \mathcal{P}_{\text{LiDAR}}} \min_{\mathbf{p} \in \mathcal{P}_{\text{GS}}} \|\mathbf{q} - \mathbf{p}\|_2,$$

where the first term penalizes geometric inaccuracies in the predicted reconstruction (accuracy), while the second term encourages coverage of the reference geometry (completeness). The scalar $\lambda$ balances the contribution of the two terms.

Chamfer distance is particularly well suited to this setting due to its robustness to large differences in point cloud density and its ability to align sparse LiDAR measurements with dense Gaussian-based reconstructions. Unlike Earth Mover's Distance or ICP-based losses, it does not require equal cardinality or explicit correspondences, and remains fully differentiable, enabling efficient optimization within a gradient-based training framework.

### 3.4.2 LiDAR supervision as a geometric prior

Given a predicted depth map rendered from the 3DGS model, 3D points are obtained by back-projecting depth values into camera coordinates using known intrinsics, and subsequently transformed into world coordinates using the optimized camera poses. This results in a predicted point cloud $\mathcal{P}_{\text{GS}} \subset \mathbb{R}^3$.

In parallel, the LiDAR sensor provides a sparse but metrically accurate point cloud $\mathcal{P}_{\text{LiDAR}} \subset \mathbb{R}^3$, that will be used as the geometric reference. To ensure that the loss is computed only over observable geometry, LiDAR points are filtered to retain those lying within the camera field of view, discarding points behind the camera or outside the image projection

Since the two point clouds may differ significantly in scale, spatial extent, and density, both sets are jointly normalized to distance computation. Specifically, a common centroid is subtracted and a robust, scene-dependent scale factor is applied. This normalization improves numerical stability and ensures that the Chamfer loss remains well-conditioned across scenes of varying size, without affecting the underlying geometric alignment objective.

### 3.4.3 Depth-aware weighting and training strategy

Direct computation of the Chamfer distance between all points in both point clouds is computationally expensive in large outdoor scenes. To reduce computational cost while preserving geometric coverage, a depth-aware subsampling strategy is employed.

The predicted point cloud is subsampled using a balanced scheme that combines a near-field bias with global spatial coverage. A fraction of points is selected preferentially from regions close to the camera, where depth estimates are typically more reliable, while the remaining points are sampled uniformly across the full depth range. This strategy ensures that both foreground structures and distant geometry contribute to the loss. The LiDAR point cloud is subsampled independently to a comparable size.

In addition, a depth-aware weighting is applied to the Chamfer term from predicted points to LiDAR points. Points located farther from the camera are assigned higher weights, compensating for perspective effects that would otherwise bias the optimization toward nearby geometry. This encourages consistent geometric alignment across both near-field and far-field regions, which is particularly important in large-scale outdoor environments.

#### Progressive integration during training

The LiDAR-guided Chamfer loss is introduced progressively during training using a warm-up strategy. This prevents early-stage instability that can arise when enforcing strict metric alignment before the photometric and depth-based components of the model have converged to a reasonable solution. By gradually increasing the influence of the Chamfer term, the model is first allowed to establish a coherent visual and depth-consistent reconstruction, after which explicit geometric alignment is enforced.

Overall, the proposed LiDAR-guided Chamfer loss addresses the global scale ambiguity and structural inconsistencies that persist under photometric and depth-based supervision alone. By explicitly aligning the 3DGS reconstruction with metrically accurate LiDAR measurements, the resulting scene representation becomes suitable for downstream robotic tasks that require metric consistency, such as reliable camera relocalisation, which is addressed in the following sections.

## 3.5 Camera Pose Estimation from a 3DGS

We formalize the camera pose estimation problem addressed in this chapter as follows: Given a single RGB query image and a known environment represented by a pre-trained 3DGS model, the objective is to estimate the corresponding camera pose as a 6-DoF rigid transformation in SE(3) with respect to the scene coordinate frame.

The scene representation is assumed to be fixed and available beforehand. In contrast to classical visual SLAM pipelines, this work does not address map building, loop closure, or joint optimization of structure and motion. Instead, the focus is exclusively on accurate camera pose estimation given an explicit 3D scene model.

Recent advances in Novel View Synthesis have demonstrated that learned and explicit scene representations can be leveraged for pose estimation. Iterative analysis-by-synthesis methods such as iNeRF [31] optimize the camera pose by minimizing photometric discrepancies between rendered and observed images, but typically require careful initialization and iterative optimization.

In contrast, explicit representations such as 3DGS enable alternative pose estimation strategies that exploit fast rendering and explicit geometric primitives, allowing efficient and robust camera pose estimation without iterative photometric refinement.

### 3.5.1 Place Recognition and Candidate Retrieval

Before performing camera pose estimation, a place recognition step is applied to reduce the search space and identify a set of candidate map images likely corresponding to the query location.

We adopt the place recognition pipeline proposed in [37], which relies on global image descriptors to retrieve visually similar mapped locations. For each RGB image in the global map, a global descriptor is precomputed offline.

To do this, first, a fast coarse filtering stage is used. Features are extracted from the query image using DINOv2 and aggregated into a global descriptor via the SALAD module [38]. SALAD performs learned feature aggregation over local descriptors, producing a single global embedding suitable for efficient large-scale image retrieval. This descriptor is matched against a database of stored map descriptors using cosine similarity available from FAISS library [39], retrieving the top 20 candidates.

The similarity score between the query descriptor $d_q$ and a map descriptor $d_i$ is computed as:

$$s_i = \frac{d_q \cdot d_i}{\|d_q\| \|d_i\|}$$

Once the top candidates are obtained, a second finer screening process is carried out. Features are extracted from the final three layers of DINOv2 transformer model [24] for both the query image and the top 20 candidates. These features offer a more detailed and semantically rich representations, which are then compared, again using FAISS with cosine similarity, to produce a refined list of the top 10 final candidates. By doing this, the process remains efficient without sacrificing accuracy, providing robustness to environmental variations and visual ambiguities. The impact of the candidate selection strategy on pose estimation performance is analyzed in the experimental evaluation (see Chapter 4).

### 3.5.2 Single-image camera relocalisation

The method proposed in 6DGS by Bortolon et al. [5] addresses camera pose estimation from a single RGB image given a pre-trained 3DGS representation of the scene. Unlike iterative analysis-by-synthesis approaches such as iNeRF [31], 6DGS does not require pose initialization nor iterative photometric optimization, enabling a single-shot estimation of the camera pose.

6DGS presents a radiant ellicell representation, as introduced by the authors, in which rays are emitted from the centers of the Gaussian ellipsoids that compose the 3DGS model. Each ellipsoid is divided into a set of spatial cells, referred to as ellicells, and rays are generated from the center of the ellipsoid passing through all the ellicells. Each ray is associated with appearance features derived from the corresponding Gaussian parameters and represents a candidate to intersect with the center of the target camera.

In parallel, visual features are extracted from the query image using a DINOv2 backbone. An attention-based matching mechanism is then employed to establish correspondences between ray features and image features. The most reliable ray–pixel correspondences are selected and used to estimate the camera pose as the solution of a weighted least-squares problem, resulting in a single-shot 6-DoF pose estimate.

Figure 3.5 illustrates the main components of the 6DGS pipeline, including ray generation from Gaussian primitives, feature matching between rays and image pixels, and the final pose estimation step.

Compared to iterative analysis-by-synthesis methods, 6DGS achieves significantly higher computational efficiency while maintaining competitive accuracy. The authors report improvements of up to 12% in rotational accuracy and 22% in translational accuracy on real-world scenes, operating at near real-time speeds of approximately 15 fps on consumer hardware.

In this work, the 6DGS pose estimation pipeline is used without modification. All

experiments share the same pose estimation algorithm, and only the underlying 3DGS scene is varied. This experimental design allows us to directly assess how improvements in geometric fidelity and metric consistency of the reconstructed scene influence camera pose estimation performance.

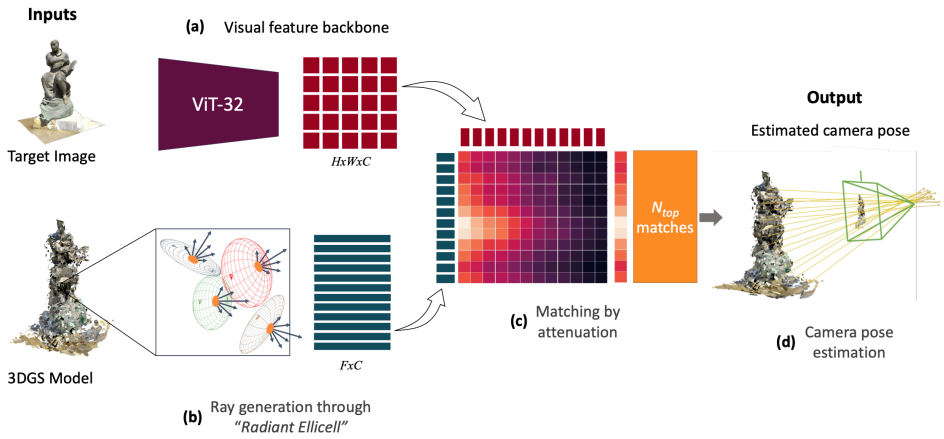



Figure 3.5: 6DGS pipeline: The image is encoded using a visual backbone (a). At the same time, rays are uniformly projected from the center of the 3DGS ellipsoids (b), and their corresponding color is estimated. Subsequently, an attention map mechanism is employed to compare the encoded ray and image features (c). Following this comparison, the N-top matches are selected via attenuation, and the camera location is estimated (d) as the solution of a weighted Least Squares problem, resulting in a distinct 6-DoF pose for the image.

# Chapter 4

# Evaluation

This chapter presents the experimental setup and a comprehensive evaluation of the proposed geometry-aware Gaussian Splatting approaches. The primary goal is to quantify the impact of depth-guided and LiDAR-guided supervision on geometric accuracy and metric scale consistency in challenging planetary-like outdoor environments. Building on this analysis, we evaluate how these geometry-aware reconstructions affect downstream camera pose estimation performance.

### 4.0.1 Dataset and sensors

Experiments are conducted on a real-world outdoor dataset collected by a mobile rover navigating large-scale, unstructured terrain. We use the **DLR Planetary Stereo Solid-State LiDAR Inertial Dataset** [40], recorded in a Moon-like volcanic environment on Mount Etna, Sicily. This dataset is specifically designed to expose the limitations of visual and LiDAR-based localisation and mapping pipelines in planetary-like scenarios.

The sensor setup includes more than ten thousand RGB frames, a solid-state LiDAR, an IMU, along with accurate D-GNSS ground truth. In this work, we use sequential monocular RGB images together with synchronized LiDAR measurements, which provide sparse but metrically accurate 3D point clouds. Camera intrinsics, approximate camera poses estimated by the SLAM system, and ground-truth information are available.

Figure 4.1 shows representative images from the dataset. As it can be seen, the scene exhibits harsh lighting conditions, extreme visual aliasing, low-texture surfaces, and a lack of salient geometric structures. These conditions significantly challenge purely photometric reconstruction methods and make visual place recognition particularly difficult. Furthermore, the LiDAR sensor exhibits a limited field of view (approximately 70° horizontal and 30° vertical), which, combined with the terrain geometry, interferes with the use of traditional LiDAR-based SLAM pipelines.

(a) Rocky slopes and unstructured volcanic terrain.

(b) Large-scale low-texture landscape with sparse landmarks.

Figure 4.1: Representative images from the DLR Planetary Stereo Solid-State LiDAR Inertial Dataset.

The dataset comprises seven sequences with diverse trajectories and terrain characteristics. Following a submap-based SLAM strategy, the global trajectory is divided into local submaps, which reflects a common practice in large-scale robotic mapping.

For each submap, the dataset provides a set of keyframe RGB images with camera poses in the local submap reference frame, and the rigid transformation relating the submap to the global coordinate system. Camera intrinsics and poses are stored in JSON format after applying the required coordinate-frame conversions, ensuring direct compatibility with the Gaussian Splatting training pipeline.

In addition, each submap includes an accurate LiDAR point cloud, which is used as a supervision signal to guide the Gaussian Splatting optimization and to enforce metric scale consistency during training.

### Coordinate frames and pose transformations

To ensure consistency between the SLAM pipeline and the Gaussian Splatting training framework, all sensor poses and geometric data are converted into a common coordinate convention compatible with Nerfstudio.

The SLAM system provides camera poses as rigid transformations from the local submap frame to the camera frame, expressed in the ROS coordinate convention (X forward, Y left, Z up). For Gaussian Splatting training, we require camera-to-world transformations following the OpenGL convention used by Nerfstudio (X right, Y up, Z backward).

Let $\mathbf{T}_{\text{submap}\to\text{cam}} \in SE(3)$ denote the pose provided by SLAM. This transform is first converted into a homogeneous matrix representation. A fixed axis-alignment

matrix $\mathbf{C}$ is then applied to convert between the ROS and OpenGL coordinate systems:

$$\mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

The final camera-to-world transformation used for Gaussian Splatting training is computed as:

$$\mathbf{T}_{\text{cam}\to\text{world}} = \mathbf{C}\,\mathbf{T}_{\text{submap}\to\text{cam}}\,\mathbf{C}^{-1}.$$

Each submap is associated with its own local coordinate frame and includes the rigid transformation relating the submap frame to the global map frame, which is stored separately to preserve metric consistency.

In addition to camera poses, each submap's aggregated LiDAR point cloud is also transformed into the same OpenGL-compatible coordinate system before being exported. This guarantees geometric alignment between photometric supervision, depth-based supervision, and LiDAR-based Chamfer loss during optimization.

## 4.1 Experimental Evaluation of Geometry-aware Gaussian Splatting

This section focuses on evaluating the impact of geometric supervision on 3D Gaussian Splatting reconstructions. We first describe the reconstruction variants considered in the experimental study, followed by the training configurations and evaluation protocol used to ensure a fair comparison.

### 4.1.1 Compared methods

The following reconstruction variants are evaluated:

**Baseline GS.** Standard Gaussian Splatting trained using photometric supervision only, without any explicit geometric constraints.

**MVSA.** Gaussian Splatting extended with depth and surface normal supervision predicted by the MVSAnywhere model, providing additional geometric regularization during training.

**MVSA + Chamfer loss.** The proposed method, which combines depth- and normal-guided supervision with a LiDAR-guided Chamfer loss to enforce global metric scale consistency.

For each reconstruction variant, experiments are conducted both with and without external point cloud priors used for Gaussian initialization.

### 4.1.2 Training configurations

All experiments are conducted using the Nerfstudio framework. Baseline Gaussian Splatting models are trained independently for each submap using the standard *splatfacto* pipeline for 30 000 iterations with the Adam optimizer. Camera intrinsics and poses are kept fixed during training, while Nerfstudio's built-in pose normalization and scene centering mechanisms are enabled. No explicit geometric supervision is used beyond photometric consistency.

Geometry-aware variants are trained using the *regsplatfacto* pipeline, which incorporates additional geometric regularization terms. In contrast to the baseline, automatic pose normalization and centering are disabled, and training is performed directly in a metric reference frame. Each submap is trained for 20 000 iterations using the Adam optimizer, with camera intrinsics and poses fixed.

Depth and surface normal supervision are provided by the MVSAnywhere model, initialized from the pretrained checkpoint `mvsanywhere_hero.ckpt`. The optimization objective combines the standard photometric loss with additional depth and normal regularization terms, weighted by $\lambda_{\text{depth}} = 0.05$ and $\lambda_{\text{normal}} = 0.1$, respectively.

For the final variant, a LiDAR-guided Chamfer loss is introduced to enforce metric scale consistency. This term is activated after 2 000 iterations and linearly warmed up until iteration 8 000, reaching a final weight of $\lambda_{\text{chamfer}} = 5 \times 10^{-5}$.

All remaining hyperparameters are kept fixed across experiments to isolate the impact of geometric supervision. A summary of the training configurations and hyperparameters for all evaluated variants is provided in Table 4.1.

### 4.1.3 Evaluation metrics

Reconstruction quality is evaluated using a combination of qualitative and quantitative metrics. Qualitatively, rendered novel views and 3D visualizations of the reconstructed geometry are analyzed to assess surface smoothness, structural coherence, and the presence of artifacts such as floating Gaussians or geometric distortions.

Quantitatively, Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM), and Learned Perceptual Image Patch Similarity (LPIPS) are used to evaluate the photometric quality of each Gaussian Splatting method. In addition, geometric consistency is evaluated using the Chamfer distance, which is defined in Section 3.4. This metric captures geometric consistency by measuring the bidirectional

| Configuration | Baseline GS | Geometry-aware GS |
|---|---|---|
| *Training setup* | | |
| Pipeline | *splatfacto* | *regsplatfacto* |
| Training iterations | 30 000 | 20 000 |
| Camera pose optimization | Disabled | Disabled |
| Pose normalization | Enabled | Disabled |
| Metric-scale training | No | Yes |
| *Geometric supervision* | | |
| Depth supervision | None | MVSAnywhere |
| Normal supervision | None | MVSAnywhere |
| LiDAR-based initialization | Optional | Optional |
| *LiDAR alignment* | | |
| Chamfer loss | Disabled | Enabled |
| Chamfer warm-up | – | 2 000–8 000 iters |
| $\lambda_{\text{depth}}$ | – | 0.05 |
| $\lambda_{\text{normal}}$ | – | 0.1 |
| $\lambda_{\text{chamfer}}$ | – | $5 \times 10^{-5}$ |

Table 4.1: Summary of training configurations for baseline and geometry-aware Gaussian Splatting variants.

distance between the reconstructed Gaussian centers and the LiDAR point cloud, penalizing both local surface inaccuracies and global misalignments.

**Peak Signal-to-Noise Ratio (PSNR)** is a pixel-wise fidelity metric that measures the similarity between a rendered image $I$ and a reference image $I^*$ based on the mean squared error (MSE). It is defined as:

$$\text{PSNR}(I, I^*) = 10 \log_{10} \left( \frac{L^2}{\text{MSE}(I, I^*)} \right), \quad \text{MSE}(I, I^*) = \frac{1}{N} \sum_{i=1}^{N} (I_i - I_i^*)^2,$$

where $L$ denotes the maximum possible pixel intensity and $N$ the number of pixels. Higher PSNR values indicate better photometric reconstruction quality.

**Structural Similarity Index Measure (SSIM)** compares local image structures in terms of luminance, contrast, and structure. Given two image patches $x$ and $y$, SSIM is defined as:

$$\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}.$$

SSIM values range between 0 and 1, with higher values indicating greater structural similarity.

**Learned Perceptual Image Patch Similarity (LPIPS)** is a perceptual metric that measures image similarity using deep feature representations extracted from a pretrained neural network. Unlike pixel-wise metrics, LPIPS captures high-level perceptual differences and correlates well with human judgments. Lower LPIPS values indicate higher perceptual similarity.

### 4.1.4 Results

#### Qualitative analysis

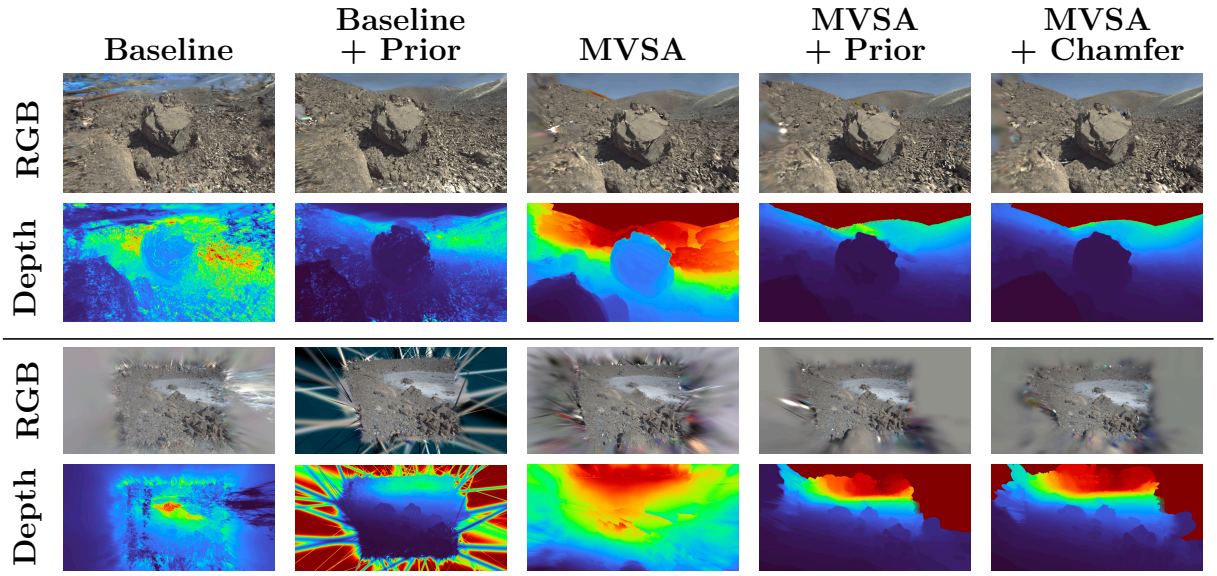



Figure 4.2: Qualitative comparison of Gaussian Splatting variants on two representative submaps. Rows correspond to RGB renderings and depth maps (blue=near, red=far), while columns compare different training configurations.

Figure 4.2 shows a qualitative comparison of the evaluated methods using both RGB renderings and the corresponding depth maps. The baseline Gaussian Splatting model struggles in these planetary-like scenes, producing blurred surfaces, loss of sharp geometric edges, and artifacts, such as floating Gaussians, especially in low-texture regions. Initializing the scene with a point cloud prior results in a more structured reconstruction and reduces artifacts, but fine details remain over-smoothed and the global geometry can still be inconsistent.

Depth- and normal-guided training with MVSAnywhere stabilizes the reconstruction by providing additional geometric constraints: depth maps become smoother and more coherent, and the recovered surfaces exhibit improved local consistency. Finally, adding LiDAR supervision through the Chamfer-based loss fixes the reconstruction to a metric reference, resulting in improved global structural consistency and more accurate depth distributions, while preserving photometric

quality in the rendered views.

Figure 4.3 illustrates a comparison between the ground truth, baseline GS, and the proposed geometry-aware approach along different submaps. Despite achieving reasonable photometric similarity, the baseline reconstruction suffers from oversmoothing and unstable appearance in weakly textured regions. The proposed method better preserves scene structure and visual consistency, leading to renderings that are more faithful to the real scene.

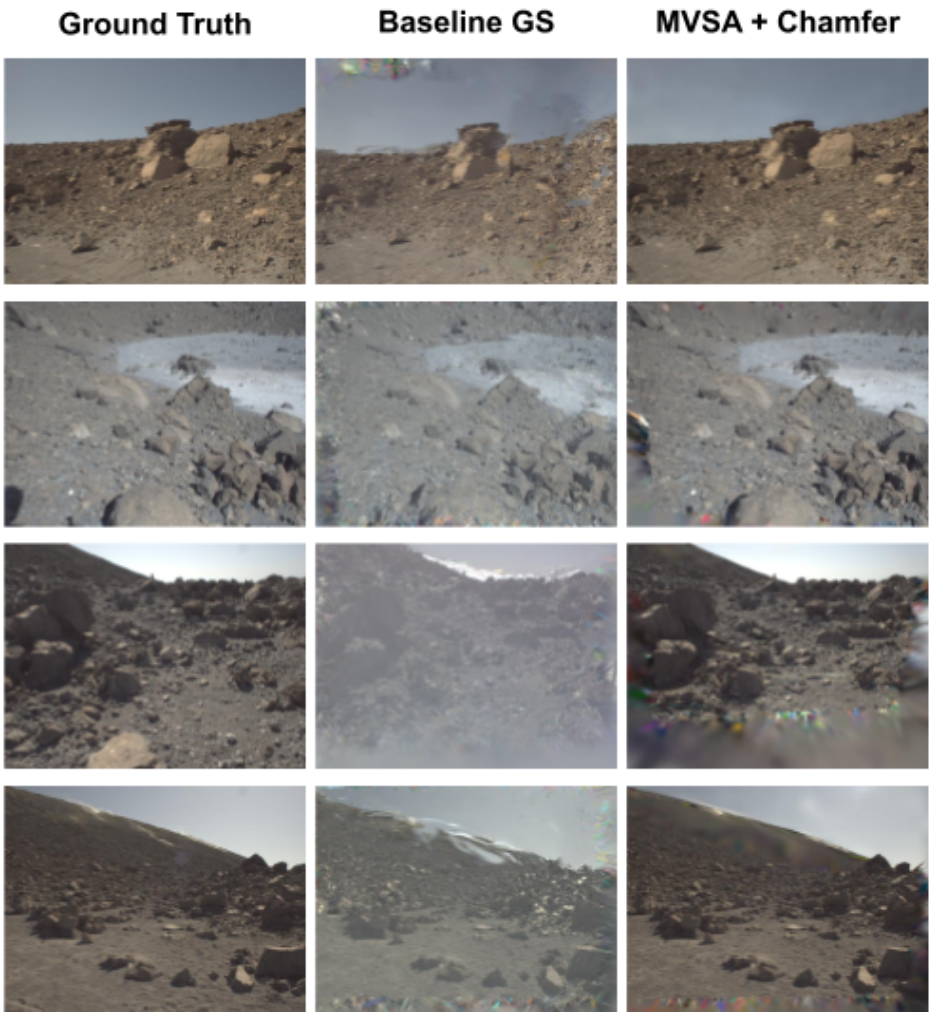



Figure 4.3: Qualitative comparison of novel view synthesis results. Each row corresponds to a different viewpoint in a planetary-like outdoor environment. From left to right: ground-truth image, baseline Gaussian Splatting, and the proposed geometry-aware method.

### Quantitative analysis

Figure 4.4 summarizes the photometric reconstruction quality and rendering performance of the evaluated methods across different submaps. The baseline Gaussian

Splatting model achieves the lowest PSNR and SSIM values, together with the highest LPIPS score, confirming the limitations of relying solely on photometric supervision.

Depth- and normal-guided supervision using MVSAnywhere alone does not improve reconstruction quality. In particular, MVSA without a point cloud prior often performs worse than the baseline. When initialized randomly, the Gaussian primitives lack a coherent spatial arrangement, and the depth predictions provided by MVSAnywhere—while locally accurate—are noisy and unreliable in regions with weak parallax and sparse viewpoints. As a result, the depth and normal losses may introduce conflicting gradients early in training, leading the optimization to converge to suboptimal local minima.

Introducing a LiDAR-derived point cloud prior as an initialization significantly improves all photometric metrics. This result highlights the importance of providing a structured geometric starting point, even when no explicit geometric supervision is applied during training.

Finally, introducing the LiDAR-guided Chamfer loss does not degrade photometric quality, maintaining high PSNR and SSIM values while preserving competitive LPIPS scores.

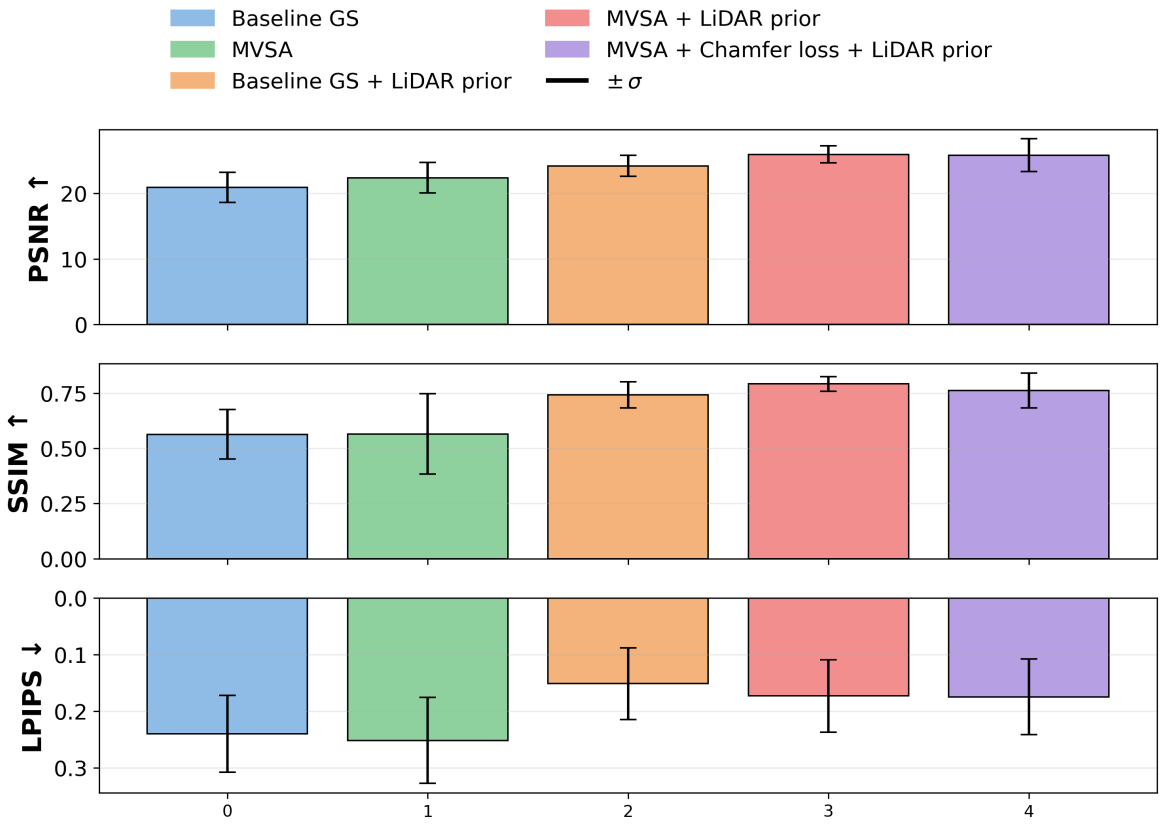



Figure 4.4: Photometric reconstruction quality averaged over all submaps. Bars show mean and standard deviation for PSNR, SSIM, and LPIPS across different seeds.

This observation is further supported by the photo-geometry trade-off analysis shown in Figure 4.5, where the proposed approach occupies the region of simultaneously high photometric quality and low geometric error.

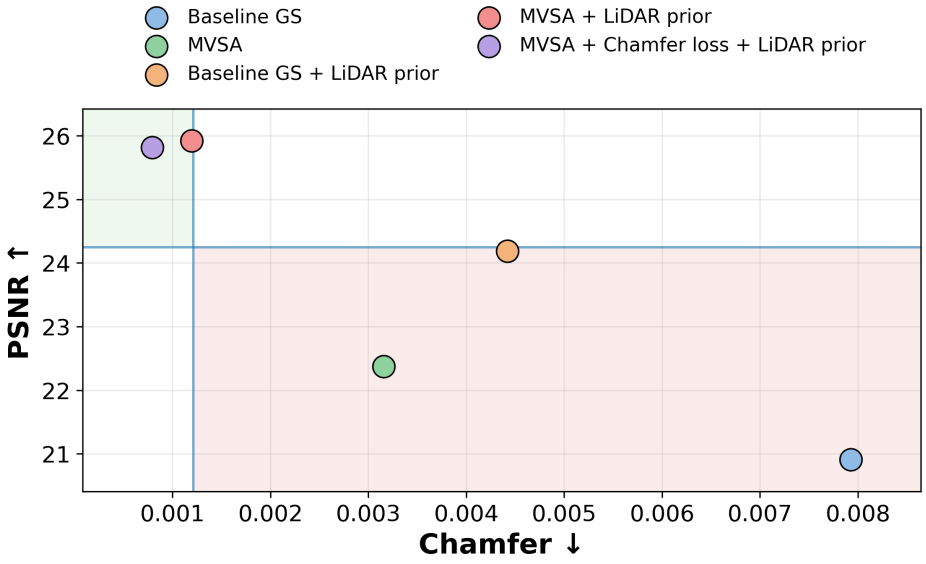



Figure 4.5: Photometric quality vs geometric accuracy across the differents cases of study.

In addition to photometric metrics, Table 4.2 reports the average Chamfer distance across different submaps for each method, while Figure 4.6 shows the distribution of errors. For clarity, percentage improvements are reported with respect to the baseline Gaussian Splatting model without geometric prior. Negative values indicate a degradation in geometric accuracy relative to the baseline.

| Method | Chamfer Distance ↓ | Improvement vs Baseline (%) |
|---|---|---|
| Baseline GS | 12.87 | – |
| MVSAnywhere | 18.14 | -41.0 |
| Baseline GS + LiDAR Prior | 1.95 | 84.9 |
| MVSAnywhere + LiDAR Prior | 1.28 | 90.1 |
| MVSAnywhere + LiDAR Prior + Chamfer Loss | **1.20** | **90.7** |

Table 4.2: Geometric accuracy measured using Chamfer distance between Gaussian Splatting reconstructions and LiDAR point clouds, averaged across all submaps.

The results reveal a clear separation between methods relying solely on photometric or depth-based supervision and those incorporating LiDAR information. LiDAR-based initialization alone reduces the Chamfer distance by nearly 85% relative to the baseline, while enforcing LiDAR supervision during training further improves geometric accuracy, achieving more than a 90% reduction with consistently low variance across submaps.

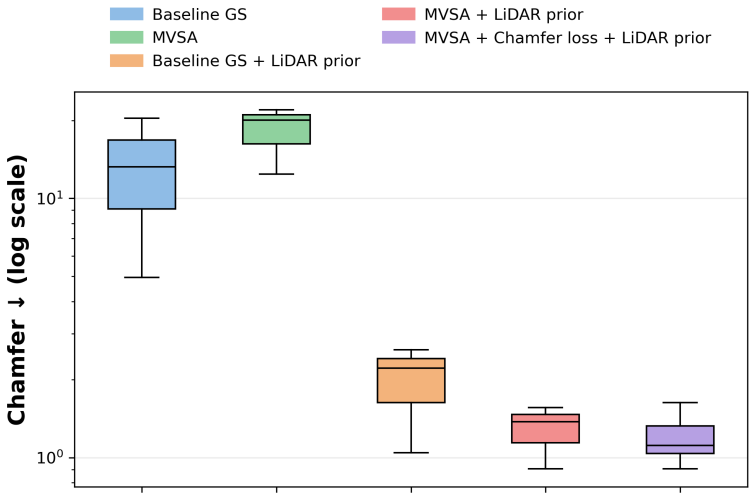



Figure 4.6: Distribution of Chamfer distances across different submaps for the evaluated methods. The logarithmic scale highlights the large performance gap between LiDAR-supervised approaches and purely photometric or depth-based methods.

Overall, these quantitative results demonstrate that enforcing metric geometric consistency through LiDAR supervision can be achieved without sacrificing visual fidelity. The proposed geometry-aware training strategy effectively mitigates the classical photo-geometry trade-off, producing reconstructions that are both visually plausible and metrically consistent.

## 4.2 Evaluation on Camera Pose Estimation

In this section, we evaluate the use of geometry-aware Gaussian Splatting reconstructions for camera pose estimation. Given a single RGB query image and a previously reconstructed scene, the goal is to estimate the full 6-DoF camera pose.

We adopt the 6DGS [5] framework, which performs camera pose estimation directly from a pre-trained 3D Gaussian Splatting model. To isolate the impact of scene geometry on relocalisation performance, the pose estimation algorithm is kept fixed while varying only the underlying Gaussian Splatting reconstruction.

### 4.2.1 Experimental setup

#### Temporal consistency with SLAM keyframes

Our first step is to retrieve the top-K candidates by performing place recognition globally in the map. To do this, we follow the approach presented in [37].

First, we restrict the image database to the set of keyframes defined by our SLAM submap construction. While the original preprocessing selected images using an appearance-based covisibility trigger (ORB matches with a median pixel-displacement

threshold), we replaced it with a timestamp-based filter that keeps only frames whose timestamps match the SLAM keyframe list (within a tolerance of 0.08 s). This ensures consistency between the retrieval database and our mapping/reconstruction pipeline. Details of the keyframe timestamp filtering are provided in Appendix A.

**Submap-level training protocol (3DGS → 6DGS)**

**Stage 1: Train 3D Gaussian Splatting per submap.** We train one Gaussian Splatting model for every submap appearing in (i) query submap and (ii) retrieved candidate submaps. This batch training is automated with the script `train_gs_top1_submaps.sh`, which iterates over a text list of submap IDs (generated from the query and retrieval outputs) and launches Nerfstudio training using the same keyframe image set employed for submap construction.

**Stage 2: Export each trained 3DGS model to the 6DGS format.** For each trained submap model, we export (i) RGB frames, (ii) camera intrinsics, (iii) camera poses, and (iv) the Gaussian model as a PLY file in the format required by 6DGS. This conversion is performed with `export_to_6dgs.py` and is orchestrated by `train_6dgs_top1_submaps.sh`.

**Stage 3: Run 6DGS pose estimation per submap.** Finally, for each exported submap, we run the 6DGS pose estimation pipeline using the same keyframe set that was used to train the corresponding 3DGS model. This ensures that relocalisation is evaluated consistently in the same temporal/keyframe regime as retrieval and reconstruction.

### 4.2.2 Evaluation metrics

The performance of the proposed relocalisation pipeline is evaluated using a set of complementary metrics that assess place recognition quality, camera pose estimation accuracy, and overall localisation robustness.

**Place recognition performance (Precision@$K$).** The quality of the image retrieval stage is evaluated using Precision@$K$, which measures the fraction of correct matches among the top-$K$ retrieved candidates for each query image. A retrieved candidate is considered correct if it corresponds to a valid ground-truth match according to the dataset annotations. Precision@$K$ therefore reflects how reliably the retrieval module ranks visually relevant locations at the top of the candidate list, independently of the pose estimation stage.

**Pose estimation accuracy.** Camera relocalisation accuracy is measured using geometric error metrics that quantify the difference between the estimated camera pose and the ground-truth pose provided by the SLAM dataset. Specifically, we report translation errors (in meters) and the yaw rotation error (in degrees).

We avoid using the camera poses optimized during Gaussian Splatting training, as these poses are adapted to minimize photometric reconstruction error and may deviate from the true scene geometry and metric scale. This choice prevents biasing the evaluation toward the reconstruction process and allows us to isolate the effect of reconstruction geometry on relocalisation performance.

**Localisation robustness (Recall@$K$).** Overall localisation performance is assessed using Recall@$K$, which measures the fraction of query images for which at least one pose hypothesis within the top-$K$ retrieved candidates satisfies a predefined geometric error threshold.

We report Recall@$K$ under two success criteria: a relaxed threshold of 10 m and 15°, which reflects coarse but usable localisation, and a stricter threshold of 2 m and 10°, corresponding to accurate metric relocalisation suitable for downstream navigation and mapping tasks.

### 4.2.3 Place recognition retrieval performance

As a first step prior to pose estimation, we evaluate the image retrieval stage used to select the top-$K$ most similar database candidates for each query image, which corresponds to the implementation used in the relocalisation pipeline at [37].

| | **Queries** | **Precision@1 ↑** | **Precision@5 ↑** | **Precision@10 ↑** |
|---|---|---|---|---|
| Retrieval | 1777 | 0.2954 | 0.2681 | 0.2633 |

Table 4.3: Place recognition retrieval performance on the evaluation set. Precision@$K$ measures the fraction of retrieved candidates within the Top-$K$ list that correspond to valid ground-truth matches for each query.

Overall, the retrieval stage achieves a Precision@1 of 0.2954, indicating that the top-ranked candidate is a valid match for approximately 29.5% of the queries. Precision decreases slightly when considering larger candidate sets (Top-5 and Top-10), which is expected in this dataset due to strong perceptual aliasing, low-texture terrain, and repeated visual patterns. These results motivate the subsequent pose estimation stage, which aims to disambiguate visually similar candidates using scene geometry.

### 4.2.4 Camera Relocalisation Results

This section evaluates camera relocalisation performance using geometry-aware Gaussian Splatting reconstructions. For each query image, the place recognition module retrieves a ranked list of candidate submaps. Pose estimation is performed independently for each candidate, producing one pose hypothesis per retrieved submap.

#### Localisation Robustness (Recall@K)

We first evaluate the system's ability to provide a "correct" pose under different tolerance thresholds. Table 4.4 shows that under the relaxed threshold, we achieve a recall of nearly 40% at $K = 5$, suggesting that the GS-based optimizer can effectively converge even when the initial retrieval provides a candidate slightly offset from the true position.

Under the stricter 2m/10° threshold, Recall@K remains nearly constant across K, indicating that increasing the number of retrieved candidates does not compensate for residual geometric inaccuracies in the reconstruction.

| $K$ | Recall@K (10m, 15°) (%) | Recall@K (2m, 10°) (%) |
|---|---|---|
| 1 | 34.1 | 7.3 |
| 3 | 39.5 | 7.0 |
| 5 | 40.9 | 6.8 |
| 10 | 39.1 | 6.5 |

Table 4.4: Pose estimation recall under two geometric thresholds. A query is successful if at least one candidate within the top-$K$ list meets the error criteria.

To better understand the interaction between retrieval ranking and pose estimation accuracy, Table 4.5 reports pose errors as a function of retrieval rank. Lower ranks generally correspond to lower pose errors, but accurate pose hypotheses are also observed at higher ranks. The large gap between mean and median errors highlights the presence of strong outliers, confirming that appearance-based retrieval alone is insufficient to guarantee geometric consistency.

| Rank | $t_{\text{med}}$ (m) | $t_{\text{mean}}$ (m) | $r_{\text{med}}$ (°) | $r_{\text{mean}}$ (°) |
|---|---|---|---|---|
| 1 | 2.61 | 2.87 | 22.67 | 26.65 |
| 2 | 2.18 | 2.29 | 20.48 | 39.80 |
| 3 | 2.84 | 2.94 | 14.42 | 26.74 |
| 4 | 3.36 | 3.27 | 24.31 | 28.97 |
| 5 | 2.70 | 2.89 | 16.30 | 24.48 |

Table 4.5: Pose estimation accuracy as a function of retrieval rank. Accurate pose hypotheses are not limited to the top-ranked retrieval result.

An important observation is that the pose hypothesis with the lowest geometric error does not necessarily belong to the ground-truth submap of the query. In several cases, accurate relocalisation is achieved using a candidate from a different submap, indicating successful geometric revisitation across submap boundaries.

Figure 4.7 provides qualitative relocalisation results. Although the rendered views often exhibit strong photometric artifacts, this behavior is expected, as the 6DGS framework does not optimize camera pose through image reconstruction. Instead, pose estimation relies on sparse geometric ray correspondences rather than full Gaussian rasterization, meaning that visual fidelity is not necessarily correlated with pose accuracy. Nevertheless, the preserved geometric structure is sufficient to enable reliable relocalisation.

The relocalisation examples shown are randomly selected and include both successful and failed pose estimations. The figure is therefore not intended to showcase best-case results, but rather to illustrate the typical behavior of the pose estimation pipeline under challenging conditions. This mixed selection highlights that, even when pose estimation fails or converges to suboptimal solutions, the reconstructed geometry often remains partially consistent, while photometric artifacts dominate the visual appearance of the renders.

### Comparison with Baseline Gaussian Splatting

We compare the camera pose estimation performance obtained with the proposed geometry-aware Gaussian Splatting representation against the baseline GS. Both approaches rely on the same place recognition pipeline and the same 6DGS pose estimation method; therefore, any observed performance differences can be directly attributed to the quality and metric consistency of the 3DGS scene representation.

### Pose Recall under Geometric Thresholds

Table 4.6 reports the pose estimation recall under two geometric accuracy thresholds for both methods. The baseline GS model exhibits very limited pose recall, even under the $(10\,\mathrm{m}, 15^\circ)$ threshold, and completely fails to recover accurate poses under stricter geometric constraints. In contrast, the proposed method achieves a substantial increase in recall across all values of $K$, being able to estimate poses with lower errors from the retrieval candidates.

To further analyze the interaction between retrieval ranking and pose estimation accuracy, Table 4.7 compares pose errors as a function of the retrieval rank for both methods.

| $K$ | Method | Recall@(10 m, 15°) [%] | Recall@(2 m, 10°) [%] |
|---|---|---|---|
| 1 | Baseline GS | 7.7 | 0.0 |
| | Ours | **34.1** | **7.3** |
| 3 | Baseline GS | 6.7 | 0.0 |
| | Ours | **39.5** | **7.0** |
| 5 | Baseline GS | 6.3 | 0.0 |
| | Ours | **40.9** | **6.8** |
| 10 | Baseline GS | 6.3 | 0.0 |
| | Ours | **39.1** | **6.5** |

Table 4.6: Pose estimation recall comparison between baseline Gaussian Splatting and the proposed geometry-aware method.

While lower retrieval ranks generally correspond to more accurate pose estimates, the baseline GS model exhibits consistently high rotation errors across all ranks, often exceeding 50°. This behavior reveals the presence of strong geometric outliers and confirms that appearance-based retrieval alone is insufficient to guarantee accurate pose estimation.

In contrast, the proposed geometry-aware representation yields significantly lower rotation errors and more stable translation estimates across retrieval ranks. Accurate pose hypotheses are not limited to the top-ranked candidate, indicating increased robustness to perceptual aliasing and retrieval ambiguity.

| Rank | Method | $t_{\text{med}}$ [m] | $r_{\text{med}}$ [°] |
|---|---|---|---|
| 1 | Baseline GS | 2.42 | 48.06 |
| | Ours | **2.61** | **22.67** |
| 2 | Baseline GS | 2.94 | 64.15 |
| | Ours | **2.18** | **20.48** |
| 3 | Baseline GS | 1.71 | 52.82 |
| | Ours | **2.84** | **14.42** |
| 4 | Baseline GS | 3.13 | 76.17 |
| | Ours | **3.36** | **24.31** |
| 5 | Baseline GS | 1.65 | 159.13 |
| | Ours | **2.70** | **16.30** |

Table 4.7: Pose estimation accuracy as a function of retrieval rank for baseline Gaussian Splatting and the proposed method.

## 4.3 Discussion

This chapter analysed geometry-aware Gaussian Splatting under planetary-like outdoor conditions and its impact on downstream camera relocalisation. The results show that photometric supervision alone is insufficient to obtain metrically consistent

reconstructions in sparse-view, low-texture environments, and that LiDAR information is the key factor to enforce global geometric consistency.

Qualitative and quantitative evaluations confirm that baseline 3DGS suffers from over-smoothed surfaces and floating artifacts, while LiDAR-based initialization significantly improves reconstruction stability and geometric accuracy. Depth- and normal-guided supervision with MVSAnywhere improves local coherence when combined with a geometric prior, but MVSA alone does not consistently improve performance and may even degrade results due to unreliable depth estimates under weak parallax and near-linear trajectories.

In contrast, LiDAR-guided Chamfer supervision anchors the reconstruction to a metric reference, yielding the lowest Chamfer distances with low variance across submaps while maintaining competitive photometric quality. As a result, the proposed method effectively mitigates the photo-geometry trade-off.

For relocalisation, place recognition performance is limited by perceptual aliasing, but geometry-aware pose estimation can recover usable poses under relaxed thresholds. Under strict thresholds, performance saturates, indicating that residual geometric inaccuracies remain the main limiting factor. Rank-wise analysis shows that accurate pose hypotheses are not restricted to the top-ranked candidate, although strong outliers persist.

Overall, the comparison demonstrates that the main limitation of baseline Gaussian Splatting for camera relocalisation lies in its lack of geometric consistency. By enforcing explicit geometric supervision during scene reconstruction, the proposed method provides a metrically reliable representation that enables 6-DoF camera pose estimation from a single RGB image, even in sparse, low-texture, and highly aliased outdoor environments.

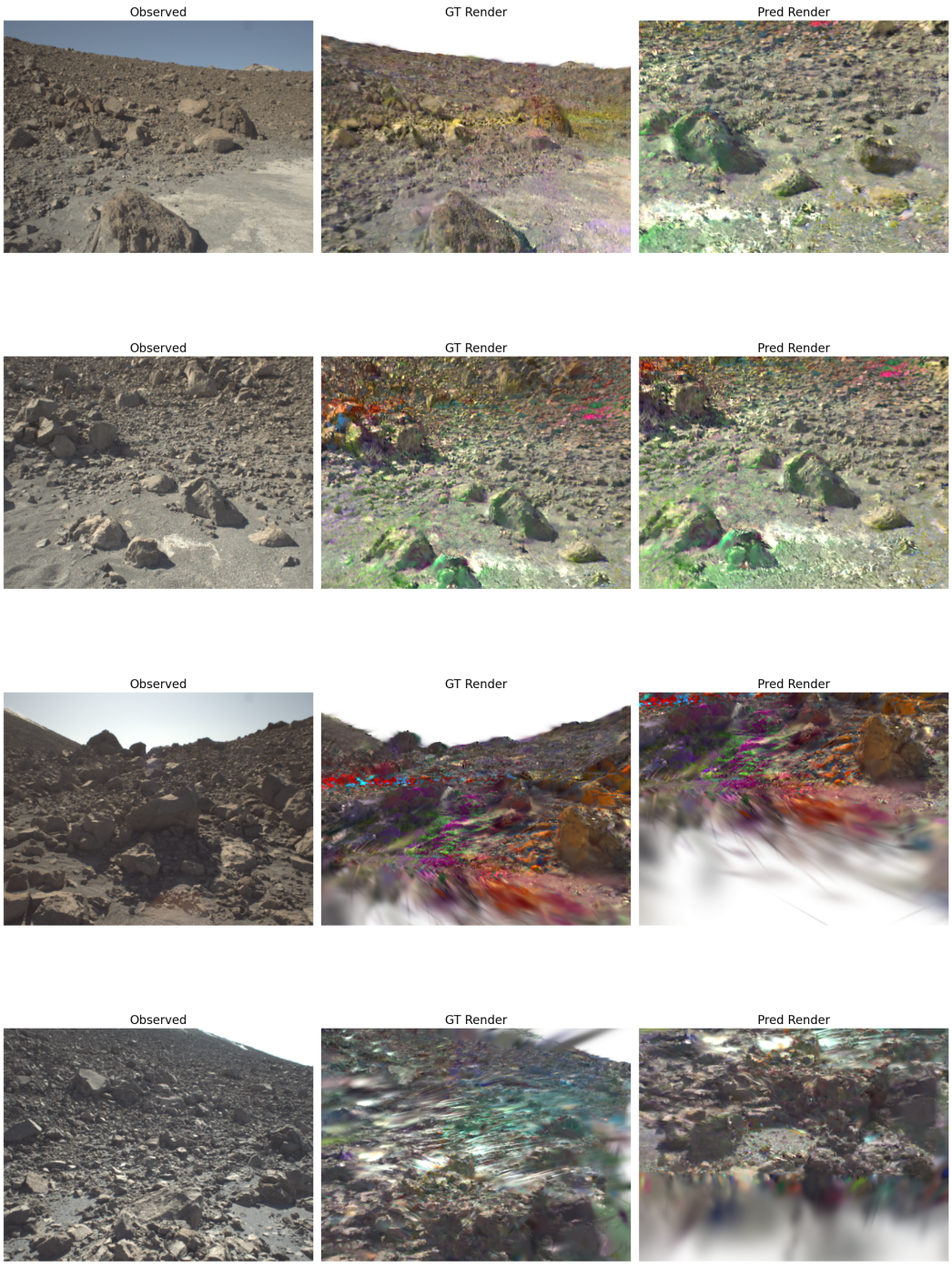



Figure 4.7: Qualitative relocalisation examples. Each row corresponds to a different query. From left to right within each image: observed image, ground-truth render, and render obtained from the estimated pose.

# Chapter 5

# Conclusions

## 5.1 Summary

This thesis addressed visual relocalisation in challenging outdoor and planetary-like environments, where low-texture surfaces, perceptual aliasing, and sparse viewpoints limit the performance of classical feature-based localisation methods. To overcome these limitations, Novel View Synthesis techniques were explored, focusing on 3D Gaussian Splatting (3DGS) as an explicit scene representation to support camera relocalisation.

An analysis of baseline Gaussian Splatting revealed that purely photometric supervision leads to geometrically inconsistent and metrically unreliable reconstructions in such scenarios. To address this, a geometry-aware framework was proposed, combining point cloud-based initialization, depth and surface normal supervision, and a LiDAR-guided Chamfer-based geometric constraint to enforce structural fidelity and global scale consistency.

Experiments conducted on a real-world planetary rover dataset demonstrate that the proposed approach significantly improves reconstruction geometry, while preserving photometric quality. These geometry-aware reconstructions result in more accurate and robust single-image 6-DoF camera relocalisation, highlighting the critical role of geometric consistency for reliable visual localisation in extreme outdoor environments

## 5.2 Challenges and Limitations

Despite these results, several limitations remain. The proposed framework relies on LiDAR measurements to enforce metric scale consistency, which may not be available in all robotic platforms. In addition, depth-guided supervision based on multi-view depth estimation is sensitive to viewpoint sparsity and initialization quality, and may be ineffective or unstable when applied without a structured geometric prior.

Furthermore, geometry-aware Gaussian Splatting introduces additional computational cost during training, which may limit scalability to very large environments or frequent map updates. Finally, relocalisation performance was evaluated using a fixed pose estimation framework, and alternative localisation strategies were not explored.

## 5.3 Future work

While the proposed framework demonstrates that explicit geometric supervision is key to reliable relocalisation in planetary-like environments, several open challenges remain and motivate future research directions.

A first promising direction is the reduction of the reliance on LiDAR supervision. Although LiDAR-based Chamfer constraints proved highly effective for enforcing metric consistency, their availability cannot be assumed in all deployment scenarios. Future work could explore the use of self-supervised geometric cues, such as cross-view depth consistency, temporal constraints along rover trajectories, or learned priors trained on planetary-scale data, as alternatives to explicit range measurements.

Additionally, the current framework operates in an offline reconstruction setting, where Gaussian Splatting models are trained independently for fixed submaps. Extending this approach to an online or incremental SLAM formulation represents an important step towards long-term autonomy. In this context, challenges such as map growth, Gaussian management, and drift-aware geometric regularisation would need to be addressed.

Finally, tighter integration between geometry-aware Gaussian representations and pose estimation remains largely unexplored. Joint optimisation strategies that couple scene reconstruction and camera relocalisation, or that explicitly reason about geometric uncertainty during pose inference, could further improve robustness in highly aliased and weakly textured environments.

# Chapter 6

# Bibliography


[1] Cesar Cadena, Luca Carlone, Henry Carrillo, Yasir Latif, Davide Scaramuzza, José Neira, Ian Reid, and John J. Leonard. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Transactions on Robotics*, 32(6):1309–1332, 2016.

[2] Carlos Campos, Richard Elvira, Juan J Gómez Rodríguez, José MM Montiel, and Juan D Tardós. Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam. *IEEE transactions on robotics*, 37(6):1874–1890, 2021.

[3] Stefan Schubert, Peer Neubert, Sourav Garg, Michael Milford, and Tobias Fischer. Visual place recognition: A tutorial [tutorial]. *IEEE Robotics & Automation Magazine*, 31(3):139–153, 2024.

[4] Riccardo Giubilato, Wolfgang Sturzl, Armin Wedler, and Rudolph Triebel. Challenges of slam in extremely unstructured environments: The dlr planetary stereo, solid-state lidar, inertial dataset. *IEEE Robotics and Automation Letters*, 7(4):8721–8728, October 2022.

[5] Matteo Bortolon, Theodore Tsesmelis, Stuart James, Fabio Poiesi, and Alessio Del Bue. 6dgs: 6d pose estimation from a single image and a 3d gaussian splatting model, 2024.

[6] A. Tewari, J. Thies, B. Mildenhall, P. Srinivasan, E. Tretschk, W. Yifan, C. Lassner, V. Sitzmann, R. Martin-Brualla, S. Lombardi, T. Simon, C. Theobalt, M. Nießner, J. T. Barron, G. Wetzstein, M. Zollhöfer, and V. Golyanik. Advances in neural rendering. *Computer Graphics Forum*, 41(2):703–735, 2022.

[7] Shenchang Eric Chen and Lance Williams. View interpolation for image synthesis. In *Proceedings of the 20th Annual Conference on Computer Graphics and*

*Interactive Techniques*, SIGGRAPH '93, page 279–288, New York, NY, USA, 1993. Association for Computing Machinery.

[8] Steven M. Seitz and Charles R. Dyer. View morphing. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '96, page 21–30, New York, NY, USA, 1996. Association for Computing Machinery.

[9] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *CoRR*, abs/2003.08934, 2020.

[10] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering, 2023.

[11] Siting Zhu, Guangming Wang, Dezhi Kong, and Hesheng Wang. 3d gaussian splatting in robotics: A survey, 10 2024.

[12] Haixing Shang, Mengyu Chen, Kenan Feng, Shiyuan Li, Zhiyuan Zhang, Songhua Xu, Chaofeng Ren, and Jiangbo Xi. Enhanced 3d gaussian splatting for real-scene reconstruction via depth priors, adaptive densification, and denoising. *Sensors*, 25(22), 2025.

[13] Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2d gaussian splatting for geometrically accurate radiance fields. In *SIGGRAPH 2024 Conference Papers*. Association for Computing Machinery, 2024.

[14] Haofei Xu, Songyou Peng, Fangjinhua Wang, Hermann Blum, Daniel Barath, Andreas Geiger, and Marc Pollefeys. Depthsplat: Connecting gaussian splatting and depth. In *CVPR*, 2025.

[15] Chi Yan, Delin Qu, Dan Xu, Bin Zhao, Zhigang Wang, Dong Wang, and Xuelong Li. Gs-slam: Dense visual slam with 3d gaussian splatting. In *CVPR*, 2024.

[16] David G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110, November 2004.

[17] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: An efficient alternative to sift or surf. In *Proceedings of the 2011 International Conference on Computer Vision*, ICCV '11, page 2564–2571, USA, 2011. IEEE Computer Society.

[18] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. *CoRR*, abs/1712.07629, 2017.

[19] Dorian Galvez-López and Juan D. Tardos. Bags of binary words for fast place recognition in image sequences. *IEEE Transactions on Robotics*, 28(5):1188–1197, 2012.

[20] Relja Arandjelovic, Petr Gronát, Akihiko Torii, Tomás Pajdla, and Josef Sivic. Netvlad: CNN architecture for weakly supervised place recognition. *CoRR*, abs/1511.07247, 2015.

[21] Gabriele Berton, Carlo Masone, and Barbara Caputo. Rethinking visual geo-localization for large-scale applications. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4878–4888, June 2022.

[22] Amar Ali-bey, Brahim Chaib-draa, and Philippe Giguère. MixVPR: Feature mixing for visual place recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2998–3007, 2023.

[23] Nikhil Keetha, Avneesh Mishra, Jay Karhade, Krishna Murthy Jatavallabhula, Sebastian Scherer, Madhava Krishna, and Sourav Garg. Anyloc: Towards universal visual place recognition. *IEEE Robotics and Automation Letters*, 2023.

[24] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2024.

[25] Paul-Edouard Sarlin, Cesar Cadena, Roland Siegwart, and Marcin Dymczyk. From coarse to fine: Robust hierarchical localization at large scale. *CoRR*, abs/1812.03506, 2018.

[26] Vincent Lepetit, Francesc Moreno-Noguer, and Pascal Fua. Epnp: An accurate o(n) solution to the pnp problem. *Int. J. Comput. Vision*, 81(2):155–166, February 2009.

[27] Martin A. Fischler and Robert C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, June 1981.

[28] Johannes L. Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[29] Paul J. Besl and Neil D. McKay. A method for registration of 3-d shapes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(2):239–256, February 1992.

[30] Alex Kendall, Matthew Grimes, and Roberto Cipolla. Convolutional networks for real-time 6-dof camera relocalization. *CoRR*, abs/1505.07427, 2015.

[31] Lin Yen-Chen, Pete Florence, Jonathan T. Barron, Alberto Rodriguez, Phillip Isola, and Tsung-Yi Lin. iNeRF: Inverting neural radiance fields for pose estimation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021.

[32] Sergio Izquierdo, Mohamed Sayed, Michael Firman, Guillermo Garcia-Hernando, Daniyar Turmukhambetov, Javier Civera, Oisin Mac Aodha, Gabriel J. Brostow, and Jamie Watson. MVSAnywhere: Zero shot multi-view stereo. In *CVPR*, 2025.

[33] Hanlin Chen, Fangyin Wei, Chen Li, Tianxin Huang, Yunsong Wang, and Gim Hee Lee. Vcr-gaus: View consistent depth-normal regularizer for gaussian surface reconstruction. *arXiv preprint arXiv:2406.05774*, 2024.

[34] Matias Turkulainen, Xuqian Ren, Iaroslav Melekhov, Otto Seiskari, Esa Rahtu, and Juho Kannala. Dn-splatter: Depth and normal priors for gaussian splatting and meshing. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2025.

[35] Junzhe Jiang, Chun Gu, Yurui Chen, and Li Zhang. Gs-lidar: Generating realistic lidar point clouds with panoramic gaussian splatting, 2025.

[36] Georg Hess, Carl Lindström, Maryam Fatemi, Christoffer Petersson, and Lennart Svensson. Splatad: Real-time lidar and camera rendering with 3d gaussian splatting for autonomous driving, 2025.

[37] Laura Alejandra Encinar González. 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, 2025.

[38] Sergio Izquierdo and Javier Civera. Optimal transport aggregation for visual place recognition, 2024.

[39] Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. The faiss library. 2024.

[40] Riccardo Giubilato, Wolfgang Stürzl, Armin Wedler, and Rudolph Triebel. Challenges of slam in extremely unstructured environments: the dlr planetary stereo, solid-state lidar, inertial dataset. *IEEE Robotics and Automation Letters*, pages 1–8, 2022.

# Appendix A

# Dataset preprocessing for Place Recognition

This appendix summarizes the preprocessing pipeline used to construct the dataset employed in the place recognition stage. Dataset preparation is based on the *S3LI Toolkit*, which provides scripts to synchronize camera images, LiDAR scans, SLAM poses, and D-GNSS measurements from the raw S3LI ROS bagfiles [40].

## A.1 Preprocessing pipeline

Dataset preparation follows a two-step procedure:

1. **SLAM pose conversion.** Camera poses estimated by a visual–inertial SLAM system are converted into time-aligned pandas dataframes using the script `slam_poses_to_pandas.py`. For each timestamp, the dataframe stores the camera position and orientation (quaternion) in a metric reference frame.

2. **Sample generation.** The script `create_dataset.py` synchronizes left camera images, LiDAR scans, SLAM poses, and D-GNSS measurements to generate the final place recognition samples.

The commands used in our experiments are:

```
python3 scripts/slam_poses_to_pandas.py <dataset_path> BASALT_STEREO_INERTIAL
python3 scripts/create_dataset.py <dataset_path> <camera_config.yaml>
```

Each generated sample consists of a synchronized tuple:

$$(I_{\text{left}},\ L,\ p_{\text{D-GNSS}},\ \phi_{\text{north}}),$$

where $I_{\text{left}}$ is the left camera image, $L$ is the associated LiDAR scan, $p_{\text{D-GNSS}}$ is the metric ground-truth position, and $\phi_{\text{north}}$ is an estimated camera yaw with respect to North.

## A.2 Keyframe-based image selection

By default, the S3LI Toolkit supports an appearance-based covisibility trigger to select image samples based on ORB feature displacement. In this thesis, this mechanism is disabled and replaced by a **keyframe-based timestamp filtering strategy** to ensure full consistency with the SLAM submaps used throughout the reconstruction and relocalization pipeline.

An external CSV file (`keyframe_timestamps.csv`) provides the timestamps of SLAM keyframes. Since the ROS bagfiles contain all images along the trajectory, only images whose timestamps correspond to a SLAM keyframe are retained. Specifically, an image with timestamp $t_{\text{img}}$ is kept if:

$$\min_i |t_{\text{img}} - t_{\text{kf}}^i| < \tau, \quad \tau = 0.08 \text{ s},$$

where $\{t_{\text{kf}}^i\}$ denotes the set of keyframe timestamps.

This choice enforces consistency between the place recognition database and the reconstruction pipeline.

The resulting dataset is stored per sequence as a pandas dataframe and exported to `.pkl`, `.csv`, and `.json` formats. For each sample, the dataset includes the image path, associated LiDAR point cloud, SLAM pose, estimated northing orientation, and D-GNSS position.